

Universidad Carlos III de Madrid

Escuela Politécnica Superior



Ingeniería Informática

Proyecto Fin de Carrera

Madrid, Julio 2012

*Sistema de Gestión y control de préstamo de  
libros en bibliotecas para teléfonos móviles  
Android*

**Autor:** Javier Fombellida Moreno

**Tutor:** Javier Fernández Muñoz



# Agradecimientos

Me gustaría agradecer en primer lugar a mi tutor Javier Fernández, por todo el apoyo y la gran ayuda prestada a lo largo del proyecto, debido a que he aprendido mucho gracias a sus indicaciones y sus aportaciones.

También me gustaría dar las gracias a mis padres Javier e Isabel, ya que todos estos años me han apoyado y han estado siempre en los buenos y malos momentos, y gracias a ellos no lo hubiese conseguido y hoy no estaría aquí. Gracias mi novia Rocío por su apoyo, por creer siempre en mí y por animarme cuándo más lo necesitaba. Gracias a mi tía Teresa por ayudarme en todo lo posible y hacer que haya alcanzado mi sueño. Gracias a mi hermana Valeria por aguantarme y animarme siempre que lo he necesitado. Gracias a mi abuela por creer siempre en mí y tener la confianza ciega de que lo lograría.

Quisiera dar las gracias también a mis amigos y compañeros Javier Torres y Víctor, por ser unos excelentes compañeros y amigos. Gracias a David, Mario, Irene y Javier Castrejón por compartir muy buenos momentos y experiencias. Gracias a Roberto, Martín, Aitor y en general, gracias a todos por estos años increíbles.

Por último quisiera dar las gracias a personas muy especiales que hoy no pueden estar aquí que tantas veces me han dado su apoyo y confianza como son mis abuelos Felipe y Amador, mi abuela Trinidad y mi tío Francisco a los que tanto añoro en momentos como este y que tantos ánimos y apoyos me han dado ya que han creído en mí.



# Resumen

El objetivo de este proyecto es desarrollar un sistema de control y gestión de préstamos mediante dispositivos móviles mediante la creación de un servicio web, el cuál es el encargado de controlar todo el sistema. El problema que este proyecto persigue, es modernizar los sistemas de préstamos tradicionales, de forma que un usuario pueda auto-prestarse un libro en una biblioteca con un dispositivo móvil.

A la hora de desarrollar este proyecto se ha realizado un estudio extendido de las tecnologías que se han utilizado, así como aplicaciones similares que existan, y en si mismo, se ha realizado un estudio del sistema y de toda la funcionalidad necesaria que se va a requerir para cumplir todos los objetivos del proyecto. En las siguientes páginas se van a detallar las tecnologías utilizadas así como los diferentes objetivos que se han marcado, seguido de un análisis y diseño del sistema. Por ultimo se va a detallar la forma de implementar el diseño realizado.

Por lo tanto, se plantea crear dos aplicaciones móviles para dos tipos de usuarios que son el administrador y el usuario final al que se le presta el libro. Estas aplicaciones serán manejadas por su tipo de usuario correspondiente de forma que a la hora de realizar alguna acción, cada aplicación móvil se va a poner en contacto con el servicio web realizando las acciones pertinentes, y obteniendo los datos de la base de datos, para que finalmente devuelva la respuesta a la aplicación móvil, mostrándosela al usuario.

Los datos obtenidos de la base de datos en todo caso van a ser manejados por el servicio web, de tal forma que por seguridad, no van a poder acceder directamente a la aplicación Android, ya que tendrá que verificar por la protección de datos, que el usuario que los solicita tiene acceso a ellos. Cada una de las aplicaciones Android sirve para un propósito diferente, debido a que están diseñadas especialmente para un tipo de usuario. De esta forma se han creado dos aplicaciones:

- **Aplicación administrador:** especialmente diseñada para los empleados que están destinados a gestionar la aplicación. Se pueden realizar tareas de préstamos y devolución de ejemplares, así como gestionar los libros y usuarios que están en el sistema.
- **Aplicación cliente:** es para aquellos usuarios que quieran que se les preste un libro. Esta aplicación a parte de realizar la tarea de auto-préstamo desde la cuál, un usuario puede prestarse un libro, permite realizar tareas tales como ver su perfil e historial de préstamos, así como renovar o reservar un ejemplar.

Las principales conclusiones obtenidas al realizar este proyecto son que se ha modernizado un sistema que usa mucha gente, debido al auge de los dispositivos móviles y de Internet, haciendo necesaria una aplicación para los usuarios de las bibliotecas con el fin de agilizar los préstamos de libros.

**Palabras clave:** Android, préstamo, biblioteca, escanear, cliente, administrador, aplicaciones móviles.



# Abstract

The main target of this Project is the development of a system of control and management of loans through mobile devices by creating a web service, which is responsible for controlling the entire system. The problem this project aims is to modernize the traditional loan schemes, so that a user can self-provision of the book in a library with the mere fact carry a mobile device based on Android.

When developing this project has been an extended study of the technologies that were to be used and a similar application exist, and in itself has made a study of the system and all the functionality needed to be requiring to meet all project objectives. In the following pages will detail the technology used and the different goals they have set, followed by an analysis and system design. Finally it will detail how to implement the design made and the system verification tests performed.

Therefore, we propose to create two mobile applications for two types of users are the administrator and the user you are given the book. These applications will be handled by the appropriate user type so that when some action, every mobile application is going to get in touch with the Web service by performing the appropriate action, and obtaining data from the database to finally return the response to the mobile application, appear to the user.

The data from the database in any case will be handled by the Web service, so that security will not be able to directly access the Android application, as will be verified by data protection, which asks the user who has access to them. Each Android application serves a different purpose, because they are designed specifically for one type of user. This would have created two applications:

- **Application administrator:** specially designed for employees who are used to manage the application. It is possible to carry out loans and return of copies, and manage the books and users in the system.
- **Client application:** is for users who want to give them a book. This application to perform the task of auto-loan from which a user can borrow a book, to perform tasks such as viewing your profile and loan history, and renew or reserve a copy.

The main conclusions to this project are to be modernized a system that uses a lot of people, due to the rise of mobile devices and the Internet, an application is necessary for library users in order to speed up loans books.

**Keywords:** Android, lend, library, scan, client, administrator, mobiles applications





# Índice General

1.	Introducción y objetivos .....	23
1.1.	Introducción.....	25
1.2.	Motivación.....	26
1.3.	Objetivos.....	27
1.4.	Posibles aplicaciones del sistema. ....	28
1.5.	Contenido de la memoria.....	30
2.	Estado del Arte.....	33
2.1.	Introducción.....	35
2.2.	Android.....	35
2.2.1.	Selección de la plataforma .....	35
2.2.2.	Introducción a Android .....	37
2.2.3.	Historia de Android .....	39
2.2.4.	Arquitectura de Android.....	42
2.3.	SOAP .....	44
2.3.1.	Introducción a SOAP .....	44
2.3.2.	Historia de SOAP .....	45
2.3.3.	Selección del protocolo .....	46
2.4.	.Net .....	47
2.4.1.	Introducción a .Net.....	47
2.4.2.	Historia de .NET .....	49
2.4.3.	Arquitectura de .Net .....	50
2.4.4.	Selección de la plataforma .....	51

2.5.	Internet Information Server .....	52
2.5.1.	Introducción .....	52
2.5.2.	Historia .....	53
2.5.3.	Selección de la plataforma .....	54
2.6.	SQL Express .....	55
2.6.1.	Introducción a SQL Express .....	55
2.6.2.	Selección de la base de datos .....	56
3.	Análisis del sistema .....	57
3.1.	Introducción.....	59
3.2.	Descripción General .....	59
3.2.1.	Capacidades generales.....	59
3.2.2.	Restricciones Generales .....	60
3.2.3.	Características de los usuarios.....	61
3.2.4.	Entorno operacional .....	61
3.3.	Requisitos de usuario.....	62
3.3.1.	Requisitos de Capacidad .....	63
3.3.2.	Requisitos de Restricción .....	72
3.4.	Casos de uso .....	75
3.4.1.	Diagramas de casos de uso.....	75
3.4.2.	Descripción textual.....	76
3.5.	Requisitos Software .....	86
3.5.1.	Requisitos de Software Funcional.....	87
3.5.2.	Requisitos no funcionales.....	94

<b>4.</b>	<b>Diseño .....</b>	<b>99</b>
4.1.	Arquitectura del sistema .....	101
4.1.1.	Modelo Vista Controlador (MVC) .....	103
4.1.2.	Modelo Cliente-Servidor .....	104
4.2.	Diseño detallado .....	106
4.2.1.	Diseño de interfaces de usuario .....	106
4.2.2.	Aplicaciones Android .....	110
4.2.3.	Servicio Web .....	115
4.2.4.	Base de datos .....	117
<b>5.</b>	<b>Implementación y Pruebas .....</b>	<b>119</b>
5.1.	Implementación .....	121
5.1.1.	Creación del servidor .....	122
5.1.2.	Implementación del servicio web .....	123
5.1.3.	Creación de la base de datos .....	124
5.1.4.	Implementación de las pantallas Android .....	124
5.1.5.	Gestión de las comunicaciones .....	127
5.1.6.	Servicio de correo .....	128
5.1.1.	Servicio de pruebas unitarias .....	129
5.2.	Pruebas .....	130
5.2.1.	Pruebas de aceptación .....	130
5.3.	Matrices de trazabilidad .....	136
5.3.1.	Matriz de trazabilidad RUC - RSF .....	136
5.3.2.	Matriz de trazabilidad RUR - RSNF .....	138
5.3.3.	Matriz de trazabilidad PA - RSF .....	139

6.	Conclusiones.....	141
6.1.	Conclusiones.....	143
6.2.	Líneas futuras .....	144
7.	Acrónimos y abreviaturas .....	147
8.	Referencias.....	151
	Anexo A: Planificación y presupuesto .....	155
1.	Planificación.....	157
1.1	Planificación de tareas .....	157
1.2	Diagrama de Gantt. ....	158
2.	Presupuesto.....	159
2.1	Costes de personal .....	159
2.2	Costes de Hardware y Software .....	159
2.3	Costes Total .....	160
	Anexo B: Manual de usuario .....	161
1.	Aplicación Usuario.....	163
1.1	Pantalla de inicio de sesión .....	163
1.2	Pantalla de menú principal.....	164
1.3	Pantalla de información de usuario.....	165
1.4	Pantalla de Otros servicios.....	169
2.	Aplicación administrador .....	173
2.1	Pantalla de inicio de sesión .....	173
2.2	Pantallas de Escanear .....	174
2.3	Pantallas de gestión del usuario .....	177
2.4	Pantallas de gestión del libro .....	178

Anexo C: Manual de Implantación.....	181
1.    Aplicaciones Android.....	183
2.    Servidor IIS .....	184
3.    Aplicaciones .NET .....	185
Anexo D: Material entregado .....	187
1.    Aplicaciones Android.....	189
1.1    Aplicación Android administrador .....	189
1.2    Aplicación Android usuario .....	189
2.    Solución .NET.....	189
1.1    Servicio Web.....	189
1.2    Base de datos .....	189
1.3    Servidor de correo.....	189
3.    Memoria .....	189



# Índice de Tablas

<b>Tabla 1: Versiones de Android</b>	41
<b>Tabla 2: Versiones de Android</b>	50
<b>Tabla 3: Versiones de IIS</b>	53
<b>Tabla 4: Entorno operacional del administrador</b>	61
<b>Tabla 5: Entorno operacional del cliente</b>	61
<b>Tabla 6: Plantilla de requisito de usuario</b>	62
<b>Tabla 7: RUC-01</b>	63
<b>Tabla 8: RUC-02</b>	63
<b>Tabla 9: RUC-03</b>	63
<b>Tabla 10: RUC-04</b>	63
<b>Tabla 11: RUC-05</b>	64
<b>Tabla 12: RUC-06</b>	64
<b>Tabla 13: RUC-07</b>	64
<b>Tabla 14: RUC-08</b>	64
<b>Tabla 15: RUC-09</b>	65
<b>Tabla 16: RUC-10</b>	65
<b>Tabla 17: RUC-11</b>	65
<b>Tabla 18: RUC-12</b>	65
<b>Tabla 19: RUC-13</b>	66
<b>Tabla 20: RUC-14</b>	66
<b>Tabla 21: RUC-15</b>	66
<b>Tabla 22: RUC-16</b>	67

<b>Tabla 23: RUC-17</b> .....	67
<b>Tabla 24: RUC-18</b> .....	67
<b>Tabla 25: RUC-19</b> .....	67
<b>Tabla 26: RUC-20</b> .....	68
<b>Tabla 27: RUC-21</b> .....	68
<b>Tabla 28: RUC-22</b> .....	68
<b>Tabla 29: RUC-23</b> .....	68
<b>Tabla 30: RUC-24</b> .....	69
<b>Tabla 31: RUC-25</b> .....	69
<b>Tabla 32: RUC-26</b> .....	69
<b>Tabla 33: RUC-27</b> .....	69
<b>Tabla 34: RUC-28</b> .....	70
<b>Tabla 35: RUC-29</b> .....	70
<b>Tabla 36: RUC-30</b> .....	70
<b>Tabla 37: RUC-31</b> .....	70
<b>Tabla 38: RUC-32</b> .....	71
<b>Tabla 39: RUC-33</b> .....	71
<b>Tabla 40: RUC-34</b> .....	71
<b>Tabla 41: RUC-35</b> .....	71
<b>Tabla 42: RUR-01</b> .....	72
<b>Tabla 43: RUR-02</b> .....	72
<b>Tabla 44: RUR-03</b> .....	72
<b>Tabla 45: RUR-04</b> .....	72
<b>Tabla 46: RUR-05</b> .....	73



<b>Tabla 47: RUR-06</b> .....	73
<b>Tabla 48: RUR-07</b> .....	73
<b>Tabla 49: RUR-08</b> .....	73
<b>Tabla 50: RUR-09</b> .....	74
<b>Tabla 51: RUR-10</b> .....	74
<b>Tabla 52: RUR-11</b> .....	74
<b>Tabla 53: RUR-12</b> .....	74
<b>Tabla 54: Plantilla de caso de uso</b> .....	76
<b>Tabla 55: CU-01</b> .....	78
<b>Tabla 56: CU-02</b> .....	78
<b>Tabla 57: CU-03</b> .....	79
<b>Tabla 58: CU-04</b> .....	79
<b>Tabla 59: CU-05</b> .....	80
<b>Tabla 60: CU-06</b> .....	80
<b>Tabla 61: CU-07</b> .....	81
<b>Tabla 62: CU-08</b> .....	81
<b>Tabla 63: CU-09</b> .....	82
<b>Tabla 64: CU-10</b> .....	82
<b>Tabla 65: CU-11</b> .....	83
<b>Tabla 66: CU-12</b> .....	83
<b>Tabla 67: CU-13</b> .....	84
<b>Tabla 68: CU-14</b> .....	84
<b>Tabla 69: CU-15</b> .....	84
<b>Tabla 70: CU-16</b> .....	85

<b>Tabla 71: CU-17.....</b>	<b>85</b>
<b>Tabla 72: Plantilla de los requisitos funcionales.....</b>	<b>86</b>
<b>Tabla 73: RSF-01.....</b>	<b>87</b>
<b>Tabla 74: RSF-02.....</b>	<b>87</b>
<b>Tabla 75: RSF-03.....</b>	<b>87</b>
<b>Tabla 76: RSF-04.....</b>	<b>88</b>
<b>Tabla 77: RSF-05.....</b>	<b>88</b>
<b>Tabla 78: RSF-06.....</b>	<b>88</b>
<b>Tabla 79: RSF-07.....</b>	<b>89</b>
<b>Tabla 80: RSF-08.....</b>	<b>89</b>
<b>Tabla 81: RSF-09.....</b>	<b>89</b>
<b>Tabla 82: RSF-10.....</b>	<b>90</b>
<b>Tabla 83: RSF-11.....</b>	<b>90</b>
<b>Tabla 84: RSF-12.....</b>	<b>90</b>
<b>Tabla 85: RSF-13.....</b>	<b>91</b>
<b>Tabla 86: RSF-14.....</b>	<b>91</b>
<b>Tabla 87: RSF-15.....</b>	<b>91</b>
<b>Tabla 88: RSF-16.....</b>	<b>91</b>
<b>Tabla 89: RSF-17.....</b>	<b>92</b>
<b>Tabla 90: RSF-18.....</b>	<b>92</b>
<b>Tabla 91: RSF-19.....</b>	<b>92</b>
<b>Tabla 92: RSF-20.....</b>	<b>93</b>
<b>Tabla 93: RSF-21.....</b>	<b>93</b>
<b>Tabla 94: RSF-22.....</b>	<b>93</b>

<b>Tabla 95: RSNF-01 .....</b>	<b>94</b>
<b>Tabla 96: RSNF-02 .....</b>	<b>94</b>
<b>Tabla 97: RSNF-03 .....</b>	<b>94</b>
<b>Tabla 98: RSNF-04 .....</b>	<b>95</b>
<b>Tabla 99: RSNF-05 .....</b>	<b>95</b>
<b>Tabla 100: RSNF-06 .....</b>	<b>95</b>
<b>Tabla 101: RSNF-07 .....</b>	<b>95</b>
<b>Tabla 102: RSNF-08 .....</b>	<b>96</b>
<b>Tabla 103: RSNF-09 .....</b>	<b>96</b>
<b>Tabla 104: RSNF-10 .....</b>	<b>96</b>
<b>Tabla 105: RSNF-11 .....</b>	<b>97</b>
<b>Tabla 106: RSNF-12 .....</b>	<b>97</b>
<b>Tabla 107: Plantilla de Pruebas de aceptación .....</b>	<b>130</b>
<b>Tabla 108: PA-01 .....</b>	<b>131</b>
<b>Tabla 109: PA-02 .....</b>	<b>131</b>
<b>Tabla 110: PA-03 .....</b>	<b>131</b>
<b>Tabla 111: PA-04 .....</b>	<b>131</b>
<b>Tabla 112: PA-05 .....</b>	<b>131</b>
<b>Tabla 113: PA-06 .....</b>	<b>131</b>
<b>Tabla 114: PA-07 .....</b>	<b>132</b>
<b>Tabla 115: PA-08 .....</b>	<b>132</b>
<b>Tabla 116: PA-09 .....</b>	<b>132</b>
<b>Tabla 117: PA-10 .....</b>	<b>132</b>
<b>Tabla 118: PA-11 .....</b>	<b>132</b>

<b>Tabla 119: PA-12 .....</b>	<b>133</b>
<b>Tabla 120: PA-13 .....</b>	<b>133</b>
<b>Tabla 121: PA-14 .....</b>	<b>133</b>
<b>Tabla 122: PA-15 .....</b>	<b>133</b>
<b>Tabla 123: PA-16 .....</b>	<b>133</b>
<b>Tabla 124: PA-17 .....</b>	<b>134</b>
<b>Tabla 125: PA-18 .....</b>	<b>134</b>
<b>Tabla 126: PA-19 .....</b>	<b>134</b>
<b>Tabla 127: PA-20 .....</b>	<b>135</b>
<b>Tabla 128: PA-21 .....</b>	<b>135</b>
<b>Tabla 129: PA-22 .....</b>	<b>135</b>
<b>Tabla 130: PA-23 .....</b>	<b>135</b>
<b>Tabla 131: PA-25 .....</b>	<b>135</b>
<b>Tabla 132: PA-25 .....</b>	<b>136</b>
<b>Tabla 133: Matriz d trazabilidad RC – RSF.....</b>	<b>138</b>
<b>Tabla 134: Matriz d trazabilidad RUR – RSNF .....</b>	<b>138</b>
<b>Tabla 135: Matriz d trazabilidad PA - RSF.....</b>	<b>139</b>
<b>Tabla 136: Tabla de coste de personal.....</b>	<b>159</b>
<b>Tabla 137: Tabla de coste de hardware/software.....</b>	<b>160</b>
<b>Tabla 138: Tabla de coste de hardware/software.....</b>	<b>160</b>

# Índice de Figuras

<b>Ilustración 1: iOS vs Android.....</b>	<b>36</b>
<b>Ilustración 2: Arquitectura de Android .....</b>	<b>38</b>
<b>Ilustración 3: Arquitectura de Android .....</b>	<b>42</b>
<b>Ilustración 4: Arquitectura de SOAP .....</b>	<b>45</b>
<b>Ilustración 5: Microsoft .Net .....</b>	<b>48</b>
<b>Ilustración 6: Arquitectura de .NET .....</b>	<b>50</b>
<b>Ilustración 7: Internet Information Server .....</b>	<b>52</b>
<b>Ilustración 8: IIS vs Apache .....</b>	<b>54</b>
<b>Ilustración 9: Diagrama de casos de uso - administrador .....</b>	<b>75</b>
<b>Ilustración 10: Diagrama de casos de uso – cliente .....</b>	<b>76</b>
<b>Ilustración 11: Arquitectura del sistema .....</b>	<b>102</b>
<b>Ilustración 12: Arquitectura MVC .....</b>	<b>103</b>
<b>Ilustración 13: Arquitectura Cliente-Servidor .....</b>	<b>104</b>
<b>Ilustración 14: Arquitectura del sistema completa .....</b>	<b>105</b>
<b>Ilustración 15: Diagrama de navegación – aplicación cliente.....</b>	<b>108</b>
<b>Ilustración 16: Diagrama de navegación – aplicación administrador .....</b>	<b>109</b>
<b>Ilustración 17: Diagrama de clases de la aplicación del administrador .....</b>	<b>113</b>
<b>Ilustración 18: Diagrama de clases de la aplicación del usuario .....</b>	<b>114</b>
<b>Ilustración 19: Diagrama de clases del servicio web .....</b>	<b>116</b>
<b>Ilustración 20: Base de datos .....</b>	<b>117</b>
<b>Ilustración 21: Configuración IIS .....</b>	<b>122</b>
<b>Ilustración 22: Pantalla de ejemplo de la aplicación administrador .....</b>	<b>125</b>

<b>Ilustración 23: Pantalla de ejemplo de la aplicación usuario .....</b>	<b>126</b>
<b>Ilustración 24: Descripción de tareas junto a duración .....</b>	<b>157</b>
<b>Ilustración 25: Diagrama de Gantt. ....</b>	<b>158</b>
<b>Ilustración 26: Pantalla de inicio de sesión. ....</b>	<b>163</b>
<b>Ilustración 27: Pantalla de menú .....</b>	<b>164</b>
<b>Ilustración 28: Pantalla de información de usuario. ....</b>	<b>165</b>
<b>Ilustración 29: Pantalla ver información .....</b>	<b>166</b>
<b>Ilustración 30: Pantalla ver préstamos.....</b>	<b>167</b>
<b>Ilustración 31: Pantalla ver historial .....</b>	<b>168</b>
<b>Ilustración 32: Pantalla de otros servicios.....</b>	<b>169</b>
<b>Ilustración 33: Pantalla de otros servicios.....</b>	<b>170</b>
<b>Ilustración 34: Pantalla renovar .....</b>	<b>171</b>
<b>Ilustración 35: Pantalla reservar .....</b>	<b>172</b>
<b>Ilustración 36: Pantalla de inicio de sesión. ....</b>	<b>173</b>
<b>Ilustración 37: Pantalla de Escanear. ....</b>	<b>174</b>
<b>Ilustración 38: Pantalla de selección de modo de préstamo .....</b>	<b>175</b>
<b>Ilustración 39: Pantalla de selección de modo de préstamo - escáner .....</b>	<b>176</b>
<b>Ilustración 40: Pantalla de gestión del usuario.....</b>	<b>177</b>
<b>Ilustración 41: Pantalla Añadir un usuario. ....</b>	<b>178</b>
<b>Ilustración 42: Pantalla de gestión del libro.....</b>	<b>179</b>
<b>Ilustración 43: Pantalla de consultar libro.....</b>	<b>180</b>
<b>Ilustración 44: Instalar aplicación Android.....</b>	<b>183</b>
<b>Ilustración 45: Servidor IIS.....</b>	<b>185</b>
<b>Ilustración 46: Solución Visual Studio 2010. ....</b>	<b>186</b>



# 1. Introducción y objetivos





## 1.1. Introducción

La finalidad de este capítulo es dar a conocer el contenido de este documento y que sirva como un primer encuentro con este proyecto de final de carrera que se ha realizado. Este capítulo sirve como una pequeña introducción de los conceptos que se explican en los sucesivos apartados del documento.

El objetivo de este proyecto es mejorar el problema de la gestión de libros y usuarios de una biblioteca, mediante una aplicación móvil para el personal que se encuentre realizando préstamos y devoluciones de libros, permitiendo gestionar de una forma sencilla cada uno de los préstamos. Además se permite administrar el material que se encuentra disponible, así como la gestión de cada uno de los diferentes usuarios de forma que se puedan realizar las diferentes tareas de una forma que resulte sencilla, eficaz y fácil tanto para los trabajadores como para los usuarios. También se ofrece una aplicación al usuario final de la biblioteca de forma que pueda obtener un préstamo mediante la funcionalidad del autopréstamo en la cuál dicho usuario es el que realiza el préstamo del libro, así como poder realizar renovaciones de préstamos o reservar un libro durante un periodo de tiempo.

El auge de las tecnologías móviles y de internet, han generado nuevas necesidades en la sociedad en la que vivimos hoy en día, grandes necesidades que hacen que hoy todo el mundo posea una conexión a internet de alta velocidad, así como de un Smartphone desde el cuál se pueden hacer innumerable número de cosas como estamos acostumbrados a ver. Por lo tanto con la llegada de los Smartphones han surgido nuevos sistemas operativos móviles como es el caso de Android, desde el cual se puede gestionar dicho dispositivo de tal forma, que se pueden dar solución a innumerables necesidades que han aparecido en esta última era.

Por ejemplo, con los teléfonos inteligentes se ha dado el caso de poder utilizarlo tanto de manera tradicional, como es el poder realizar llamadas a otras personas, así como estar conectado a internet en todo momento recibiendo las notificaciones de correo electrónico en tiempo real, así como estar en constante contacto a través de redes sociales con cada uno de tus contactos. Otro uso es el caso de poder tomar fotografías y vídeos de alta definición, así como de poder escanear y generar todo tipo de etiquetas, desde el código de barras, hasta los famosos códigos QR.

Es proyecto fin de carrera ofrece una solución al problema de la gestión de libros y usuarios en una biblioteca, a través de aplicaciones Android que acompañarán al usuario en todo momento, y desde las cuales el usuario puede autoprestarse un libro, así como realizar una reserva o una renovación de algún ejemplar que tenga prestado.

## 1.2. Motivación

El desarrollo tecnológico ha evolucionado en los últimos 50 años a un ritmo a veces más rápido que la sociedad a quien van dirigidas mejoras. Desde el invento de la rueda cualquier avance tecnológico está pensado para favorecer la vida de los individuos y esto requiere siempre un ajuste entre el desarrollo de la tecnología y la practicidad cotidiana.

Este proyecto fin de carrera se plantea como un intento de mejorar un servicio tradicional de préstamos, de forma que el personal de la biblioteca o el usuario cuyo fin es obtener un préstamo, les resulte más sencillo el poder obtener un préstamo debido a que únicamente tienen que hacer uso de su Smartphone.

Actualmente, en las bibliotecas se usa el sistema tradicional de préstamo de libros, que consiste en una máquina con la capacidad de poder leer los códigos de barras de los libros, se pasa por cada uno de ellos siempre que un usuario está interesado en que se pasa por el libro cada vez. Por este motivo se propone mejorar este servicio, actualizándolo mediante el uso de nuevas tecnologías, como son el caso de usar una aplicación en un Smartphone con conexión a internet.

La motivación principal de este proyecto es por lo tanto mejorar el servicio de préstamo de libros y que produzca en una mayor agilidad y eficacia tanto para el usuario como el empleado de estos servicios bibliotecarios.

También, la ventaja del sistema de gestión de objetos en una biblioteca que se propone frente a otros sistemas tradicionales surge a partir de la posibilidad de los teléfonos inteligentes de poder escanear distintos tipos de etiquetas a través de la cámara de fotos que lleva incorporado el teléfono. Esta característica permite al usuario, poder hacer una fotografía a la etiqueta del libro desde su teléfono móvil. Una vez realizado este paso, se podrá gestionar el préstamo o devolución del libro, de forma que el usuario que quiere adquirir el artículo lo obtenga de una forma más sencilla. Posteriormente el usuario administrador podrá gestionar cada uno de los libros y a otros usuarios que estén dentro del sistema.

Por último, el poder escanear con la aplicación diferentes tipos de etiquetas, se nos presenta la oportunidad de escanear los carnets de estudiantes con un tipo de etiqueta diferente dando la posibilidad de tener un escaneador múltiple de códigos inteligentes.

### 1.3. Objetivos

El objetivo fundamental del proyecto es el dar solución a los problemas anteriormente descritos. Dicho objetivo principal se define como:

- Crear una aplicación que permita realizar la gestión y el control de préstamo de libros a través del usuario final, así como gestionar a los diferentes usuarios y libros que integrarán dicho sistema.

Tomando como base ese objetivo principal, se proponen los siguientes objetivos secundarios:

- Crear un sistema que agilice el proceso tradicional de préstamo de libros en una biblioteca.
- Adquirir los conocimientos necesarios para poder implementar dicho sistema en una plataforma móvil, así como su integración con una base de datos alojada en un servidor.
- Adquirir los conocimientos necesarios para poder implementar un sistema distribuido mediante un servicio web, haciendo uso de tecnologías modernas, tales como .NET.
- Diseñar un servicio web asociado a una base de datos que contenga toda la información relativa a la gestión y control de la aplicación móvil.
- Diseñar un prototipo funcional que se pueda ejecutar en un terminal real, con el fin de poder gestionar el préstamo de libros.
- Diseñar un prototipo funcional que se pueda ejecutar en un terminal real, con el fin de poder escanear etiquetas inteligentes.
- Añadir funcionalidades que resulten interesantes y útiles para el usuario de una aplicación de este tipo.
- Diseñar los diferentes aspectos del sistema de una forma modular que hagan que esos módulos sean reutilizables.
- Presentar un sistema que sea atractivo para el cliente con una GUI (interfaz de usuario) amigable.
- Crear varias aplicaciones en distintas tecnologías que colaboren entre sí de forma que trabajen de forma paralela.

Como conclusión de los objetivos, hay que comentar que se ha creado un sistema distribuido cliente-servidor, a través de un servicio web, el cual será el encargado de gestionar la información de una base de datos alojada en un servidor, de tal forma que se cumplan cada uno de los objetivos mencionados anteriormente.

#### **1.4. Posibles aplicaciones del sistema.**

En la actualidad existen varios escenarios a los que se puede aplicar el uso de este sistema a desarrollar y que dependen de varios factores, tales como el tipo de cliente que quiera usar la aplicación, así como el presupuesto final que se destine a desarrollar el proyecto.

La aplicación más probable es la sustitución del sistema tradicional de préstamo y devolución de libros en una biblioteca por otro más moderno, a través de un teléfono móvil, aportando notables ventajas tanto para el bibliotecario, como para el usuario consumidor del préstamo. Los diferentes pasos que se seguirían para llevar a cabo esta realidad son los que se comentan a continuación:

En una biblioteca se tiene un sistema de control y gestión de préstamos y devoluciones que se realizan a través de un teléfono móvil, de forma que este dispositivo sea capaz de escanear el código de barras del artículo, obteniendo de esta forma toda la información necesaria para llevar a cabo el préstamo del libro. Para la lectura de la etiqueta, hace falta un teléfono móvil que contenga la instalación de esta aplicación desarrollada, y a su vez, dicho teléfono contiene el sistema operativo Android, de forma que a través de la cámara de fotos del mismo, sea capaz de leer el código de barras y a través de una conexión a Internet, poder obtener los datos del libro que se quiere prestar.

- La biblioteca también puede gestionar a cada uno de los usuarios pudiendo ver cada uno de los préstamos que tiene en vigor en el momento actual de la consulta, así como poder gestionar a cada usuario dándolo de alta o baja en el sistema bibliotecario.
- La biblioteca se beneficiará del sistema implementado debido, a que no es necesaria una estación fija con el sistema tradicional de préstamos, sino que con este nuevo sistema, únicamente hace falta un dispositivo con la aplicación instalada de forma que este pueda gestionar los préstamos y devoluciones desde cualquier lugar.
- Para hacer la experiencia del préstamo más completo, no es necesario tener un único punto de préstamo al que acudirían los usuarios, sino que cada uno de los bibliotecarios podrá tener la aplicación desarrollada en un dispositivo propio.

- Cada uno de los usuarios consumidores del préstamo, podrán autoprestarse los libros. Únicamente deberán tener un teléfono inteligente con la aplicación instalada en el. De esta forma se benefician también de las ventajas del sistema, de forma que no se necesiten intermediarios a la hora de realizar un préstamo y por tanto no tienen que esperar excesivas colas a la hora de realizar la gestión.
- El cliente consumidor del préstamo puede ahora renovar los préstamos que tiene de forma que si quiere pueda acceder volver a tener los préstamos aumentando el tiempo que los tenga en su poder. De la misma forma el usuario podrá reservar libros que quiera que se le presten.

Una de las preocupaciones más evidentes sobre el proceso anterior es cómo controlar que al usuario consumidor del préstamo, a la hora de abandonar la biblioteca ya que los libros contienen una alarma que hay que desactivar cuándo un usuario ha realizado un préstamo a la salida de la biblioteca. El sistema propuesto consiste en devolver un código a la hora de realizar el préstamo, con el objetivo de introducir dicho código en una máquina desactivadora del magnetismo de las etiquetas, que se habilitará si el código es correcto. En caso contrario de no obtener el código, no se puede desactivar magnéticamente el libro y por lo tanto, el usuario no se lo puede llevar.

## 1.5. Contenido de la memoria

En este apartado se va a proceder a explicar el contenido de este documento, viendo cada uno de los capítulos que nos podemos encontrar, los cuales componen este documento. Está destinado a ofrecer una breve explicación de lo que podemos encontrar en cada uno de los capítulos que componen este documento.

1. **Introducción y objetivos:** en este apartado se va a mostrar la motivación de este proyecto, así como los objetivos que se desean alcanzar.
2. **Estado del arte:** en este apartado de especial importancia se va a mostrar el problema principal del proyecto que se pretende resolver. Además se mostrará una visión de las tecnologías y aplicaciones usadas, así como de la funcionalidad del conjunto del sistema.
3. **Análisis del sistema:** en esta sección se describe y se comenta brevemente cada una de las tecnologías utilizadas. Principalmente se va a centrar de la plataforma Android sobre la que se desarrolla el grueso del proyecto, sin dejar de lado la tecnología .Net en la cuál se ha creado el servicio web.
4. **Diseño del sistema:** en este apartado se va a realizar un estudio del sistema a desarrollar. El diseño mostrado es de forma global, aunque se puede ver de forma más completa los detalles arquitectónicos de los distintos módulos que lo conforman.
5. **Implementación:** en esta sección se pueden ver los diferentes detalles de cada uno de los módulos que contiene el sistema desde el punto de vista de la implementación. Se describen las diferentes estrategias seguidas a la hora de implementar el sistema así como las dificultades surgidas.
6. **Conclusiones:** en este apartado se especificarán los problemas más importantes que se han encontrado a la hora de desarrollar el proyecto, con una breve conclusión. A su vez en este apartado se van a mostrar las tendencias que puede tener el sistema al finalizar este proyecto. Se dará una visión global que conducirá el camino a un futuro desarrollo.
7. **Referencias y abreviaturas:** se describen cada una de las referencias y abreviaturas mencionadas en alguno de los capítulos del documento.
8. **Bibliografía:** en esta sección se van a ilustrar todo el material consultado a la hora de realizar el presente documento.

Además se incluyen una serie de anexos para la correcta utilización y comprensión del sistema.

**Anexo A – Planificación y presupuesto:** se desglosa un presupuesto con todos y cada uno de los elementos que han sido necesarios a la hora de desarrollar el proyecto.

**Anexo B – Manual de usuario:** se crea un manual de usuario de cada una de las dos aplicaciones Android, de forma que al usuario final del sistema le resulte fácil de usar en el caso de que tenga problemas con alguna de sus funcionalidades.

**Anexo C – Manual de implantación:** se crea un manual de cómo se despliega todo el sistema de forma que un usuario pueda instalarse cada una de las aplicaciones así como del servidor.

**Anexo D – Material entregado:** lista de todos los módulos entregados, así como un manual para su instalación y puesta en marcha de una forma correcta.







## 2. Estado del Arte



## 2.1. Introducción

En este capítulo se presenta un estudio de las diferentes tecnologías existentes hasta el momento con respecto al sistema a construir. De esta forma se realiza un estudio de las tecnologías que hay hasta el momento, con el fin de obtener un producto final de calidad.

Además en este capítulo se van a analizar productos similares que existen en el mercado comentando sus características, y finalmente realizando una comparativa entre todos ellos, analizando las ventajas que aportan cada uno de ellos en función con el sistema a desarrollar en este producto.

El objetivo final de este estudio es obtener una mayor visión del entorno tecnológico y así valorar otros productos similares evaluando si el producto desarrollado es viable en la actualidad.

## 2.2. Android

Android es un Sistema Operativo además de una plataforma de Software basada en el kernel de Linux. Diseñada en un principio para dispositivos móviles, Android permite controlar dispositivos por medio de bibliotecas desarrolladas o adaptadas por Google mediante el lenguaje de programación Java. Por lo tanto se puede decir que Android es una plataforma de código abierto. Esto quiere decir, que cualquier desarrollador puede crear y desarrollar aplicaciones escritas con lenguaje C u otros lenguajes y compilarlas a código nativo de ARM.

### 2.2.1. Selección de la plataforma

A la hora de realizar el proyecto, se han tenido que tomar una serie de decisiones sobre las plataformas disponibles en el mercado de los teléfonos móviles que se ajustase a las necesidades que se tenían en torno al Software y Hardware del proyecto.

En todo momento se pensó en dos plataformas móviles, *iOS* y *Android*, para el desarrollo del proyecto debido a su sencillez, madurez e integración en los mercados. Ambas plataformas se integran bastante bien con todo tipo de comunicaciones y bases de datos, por lo que cualquiera de las dos inicialmente sirve para el desarrollo del sistema e integrarlo así con el servicio web y la base de datos. Por estas razones se han destacado de entre todas las plataformas móviles, estas dos, y no se han elegido otras como son *Windows Phone 7*, *Bada* o *Webos*. Otra razón por la que se han destacado estas dos plataformas móviles de entre todas las plataformas mencionadas, es por el mero hecho de que son las más atractivas para el usuario, y de cara a la continuidad en un futuro, del desarrollo del proyecto con mayores características y funcionalidades, es necesario usar una plataforma que este al alcance del mayor número de usuarios posible, ya que de otra forma, el sistema creado no es atractivo públicamente y por lo tanto caería en el olvido. Las características de *iOS* y *Android* que hacen que sea atractivo, son básicamente que son dispositivos con interfaces táctiles, con unas capacidades de Hardware bastante avanzadas, con procesadores potentes que incluso por dispositivo

alcanzan los 4 núcleos, cosa que da mucha potencia, haciendo que se desarrollen aplicaciones muy potentes. El problema que tiene *iOS* en comparación con Android es que tan sólo se puede utilizar en un dispositivo de *Apple*, mientras que Android se utiliza en múltiples dispositivos siendo entre ellos muy competentes en cuanto a cuestiones técnicas nos podemos referir, como son los terminales de marcas como *HTC*, *Samsung* o *LG*. En cambio, una ventaja que tiene *iOS* es que a la hora de desarrollar el sistema, es mucho más homogéneo ya que al ser un único dispositivo, el diseño de la interfaz es más sencillo ya que se cuenta en todo momento con una única dimensión de pantalla, mientras que en Android, tenemos teléfonos de varias dimensiones que llegan incluso a las 5 pulgadas como es el caso del *Samsung Galaxy Note*.



**Ilustración 1: iOS vs Android**

El motivo final de desarrollar para Android ha sido por la librería principal del proyecto que es la de poder escanear códigos inteligentes. En *iOS*, esta librería se llama *ZBar* y al igual que la de *Android ZXing*, ofrece lectura de múltiples formatos de códigos inteligentes, pero la integración de la librería con el proyecto en *objective-C* para *iOS* da más problemas que su homóloga debido a que no es una librería desarrollada exclusivamente para dicha plataforma. En cambio la librería *ZXing* para Android, ha sido desarrollada por Google y se integra perfectamente con cualquier proyecto Android dando unos resultados óptimos a la hora de escanear cualquier tipo de etiqueta inteligente.

Una vez elegida la plataforma móvil que será Android por las razones expuestas anteriormente, es hora de elegir el dispositivo móvil que se va a utilizar. En este caso nos da igual uno que otro, pero por disponibilidad del desarrollador y por ser un buen dispositivo que ha dado buenos resultados en el mercado, se ha elegido el *Samsung Galaxy S* con la versión e Android 2.3 que ha día de hoy es la más estable que existe por lo que se garantiza frente a otras versiones la estabilidad del sistema.

### 2.2.2. Introducción a Android

Android es sistema operativo para móviles basado en Linux y diseño única y exclusivamente para teléfonos móviles (o Smartphones) y tablets. Además, pero no es gran medida, también hay otros dispositivos que funcionan con Android como pueden ser: relojes, reproductores de MP3, ordenadores, Netbooks e incluso televisores.

El sistema operativo Android fue adquirido por Google en Julio de 2005, pero no fue hasta el año 2007, al adquirir la compañía Open Handset Alliance, la cual es un consorcio de varias compañías líderes en tecnología móvil, entre las que se encuentran *Texas Instruments*, *Broadcom*, *Nvidia*, *Qualcomm*, *Samsung*, *Intel*, *LG*, *Motorola* y *T-Mobile*. El objetivo del consorcio respecto a Android, era el de desarrollar estándares abiertos para dispositivos móviles.

Android permite controlar dispositivos por medio de bibliotecas desarrolladas o adaptados por Google mediante el lenguaje de programación Java. Además es una plataforma de código abierto. Esto quiere decir, que cualquier desarrollador puede crear y desarrollar aplicaciones escritas con lenguaje C u otros lenguajes y compilarlas a código nativo de ARM (API de Android). El responsable del mayor despliegue del código fuente de Android bajo licencia de Software Apache ha sido Google, haciendo uso de una licencia de software libre y de código abierto a cualquier desarrollador.

El sistema permite programar aplicaciones en una variación de Java llamada Dalvik. El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, las llamadas, la agenda, etc.) de una forma muy sencilla en un lenguaje de programación muy conocido como es Java. Esta sencillez, junto a la existencia de herramientas de programación gratuitas, hace que una de las cosas más importantes de este sistema operativo sea la cantidad de aplicaciones disponibles, que extienden casi sin límites la experiencia del usuario. Una de las mejores características de este sistema operativo es que es completamente libre. Es decir, ni para programar en este sistema ni para incluirlo en un teléfono hay que pagar nada. Y esto lo hace muy popular entre fabricantes y desarrolladores, ya que los costes para lanzar un teléfono o una aplicación son muy bajos. Cualquiera puede bajarse el código fuente, inspeccionarlo, compilarlo e incluso cambiarlo. Esto da una seguridad a los usuarios, ya que algo que es abierto permite detectar fallos más rápidamente. Y también a los fabricantes, pues pueden adaptar mejor el sistema operativo a los terminales.

La estructura del sistema operativo Android se compone de aplicaciones que se ejecutan en un framework Java de aplicaciones orientadas a objetos sobre el núcleo de las bibliotecas de Java en una máquina virtual *Dalvik* con compilación en tiempo de ejecución. Las bibliotecas escritas en lenguaje C incluyen un administrador de interfaz gráfica (*surface manager*), un framework *OpenCore*, una base de datos relacional *SQLite*, una Interfaz de programación de API gráfica *OpenGL ES 2.0 3D*, un motor de renderizado *WebKit*, un motor gráfico *SGL,SSL* y una biblioteca estándar de *C Bionic*.







**Ilustración 2: Arquitectura de Android**

En cuanto a las características de Android, comentar que es un sistema operativo bastante completo con multitud de características que hacen que sea un sistema operativo sobre el que merezca la pena desarrollar aplicaciones. A la hora de desarrollar para Android, se tiene acceso completo a los API's del framework usados por las aplicaciones base, dando una gran ventaja de realizar aplicaciones de la misma calidad como las desarrolladas por Google.

La arquitectura está diseñada para simplificar la reutilización de componentes, es decir, cualquier aplicación puede publicar sus capacidades y cualquier otra aplicación puede luego hacer uso de las mismas capacidades. A la hora de desarrollar para Android, se incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de trabajo de aplicaciones de Android. Algunas de estas bibliotecas son: System C library (implementación biblioteca C estándar), bibliotecas de medios, bibliotecas de gráficos, 3D y SQLite, entre otras. Además se incluye un conjunto de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en cada una de las bibliotecas del lenguaje Java, el cual se usa para desarrollar las distintas aplicaciones de la plataforma. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual. Por último comentar que Android depende de Linux tal y como se comentó anteriormente, para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. El núcleo también actúa como una capa de abstracción entre el hardware y el software.

### 2.2.3. Historia de Android

Android a lo largo de su historia, ha desarrollado multitud de versiones, que van mejorándose una tras otra, cubriendo las necesidades de los dispositivos que se encuentran en el mercado en ese momento. Por este motivo, en este apartado se va a explicar brevemente la historia de Android, centrándome en las diferentes versiones que han existido desde que fue lanzado, viendo las mejoras que ha habido versión tras versión.

Versión	Icono	Descripción
1.0		Liberado el 23 de septiembre de 2008
1.1		Liberado el 9 de febrero de 2009
1.5 Cupcake		<p>El 30 de abril de 2009, la actualización 1.5 (Cupcake) para Android fue liberada.</p> <ul style="list-style-type: none"><li>▪ Posibilidad de grabar y reproducir videos.</li><li>▪ Capacidad de subir videos e imágenes desde el teléfono</li><li>▪ Teclado con predicción de texto</li><li>▪ Soporte para Bluetooth</li><li>▪ Nuevos widgets</li><li>▪ Transiciones de pantalla animadas</li></ul>
1.6 Donut		<p>El 15 de septiembre de 2009, el SDK 1.6 (Donut) fue liberado.</p> <ul style="list-style-type: none"><li>▪ Una interfaz integrada de cámara de fotos y video</li><li>▪ Búsqueda por voz con mayor integración con aplicaciones nativas</li><li>▪ Actualización de soporte para CDMA/EVDO, 802.1x, VPN y text-to-speech</li><li>▪ Soporte para resoluciones de pantalla WVGA</li><li>▪ Mejoras de velocidad en las aplicaciones de búsqueda y cámara</li><li>▪ Framework de gestos y herramienta de desarrollo GestureBuilder</li><li>▪ </li></ul>

### 2.1 Eclair



El 26 de octubre de 2009, el SDK 2.0 (Eclair) fue liberado.

- Velocidad de hardware optimizada
- Soporte para más tamaños de pantalla y resoluciones
- Interfaz de usuario renovada
- Soporte integrado de flash para la cámara
- Zoom digital
- Teclado virtual mejorado
- Bluetooth 2.1

### 2.2 Froyo



El 20 de mayo de 2010, el SDK 2.2 (Froyo) fue liberado.

- Optimización general del sistema Android
- Funcionalidad de Wi-Fi hotspot y tethering por USB
- Permite desactivar el tráfico de datos a través de la red del operador
- Soporte para la instalación de aplicación en la memoria expandible
- Soporte para pantallas de alto número de Puntos por pulgada, tales como 720p

### 2.3 Gingerbread



El 6 de diciembre de 2010, el SDK 2.3 (Gingerbread) fue liberado.

- Soporte nativo para telefonía VoIP SIP
- Soporte para reproducción de videos WebM/VP8 y decodificación de audio AAC
- Soporte para Near Field Communication
- Soporte mejorado para desarrollo de código nativo
- Recolección de elementos concurrentes para un mayor rendimiento
- Soporte nativo para más sensores (como giroscopios y barómetros)
- Un administrador de descargas para descargar archivos grandes
- Administración de la energía mejorada y control de aplicaciones mediante la administrador de tareas
- Cambio de sistema de archivos de YAFFS aext4



### 3.2 Honeycomb



- Sistema multitarea mejorado
- Soporte para videochat mediante Google Talk
- Mejor soporte para redes Wi-Fi,
- Añade soporte para una gran variedad de periféricos y accesorios con conexión USB: teclados, ratones, hubs, dispositivos de juego y cámaras digitales. Cuando un accesorio está conectado, el sistema busca la aplicación necesaria y ofrece su ejecución.

### 4.0 Ice Cream Sandwich



- Versión que unifica el uso en cualquier dispositivo, tanto en teléfonos, tablets, televisores, netbooks, etc.
- Aceleración por hardware, lo que significa que la interfaz podrá ser manejada y dibujada por la GPU y aumentando notablemente su rapidez, su respuesta y evidentemente, la experiencia de usuario.
- Multitarea mejorada
- Gestor del tráfico de datos de internet.
- La aplicación de la cámara mejorada
- Android Beam es la nueva característica que nos permitirá compartir contenido entre teléfonos. VíaNFC (Near Field Communication).
- Reconocimiento de voz del usuario
- Reconocimiento facial, lo que haría que puedas cambiar la vista
- Un único y nuevo framework para las aplicaciones
- Soporte nativo del contenedor MKV
- Soporte nativo para el uso de Stylus (lápiz táctil).

Tabla 1: Versiones de Android

## 2.2.4. Arquitectura de Android

Como se ha comentado anteriormente, Android es un sistema operativo diseñado por capas como se puede apreciar en la *Ilustración 2*. Anteriormente se ha comentado que la plataforma está basada en el núcleo de Linux. En concreto Android usa el kernel modificado de Linux v2.6 que versión tras versión se va actualizando hasta la versión en la que me centro en el proyecto (v2.3 de Android) que contiene la versión del kernel de Linux 2.6.35.7. En la siguiente imagen se puede apreciar la arquitectura de Android, como efectivamente es por capas, siendo comentada capa a capa a continuación de la misma.

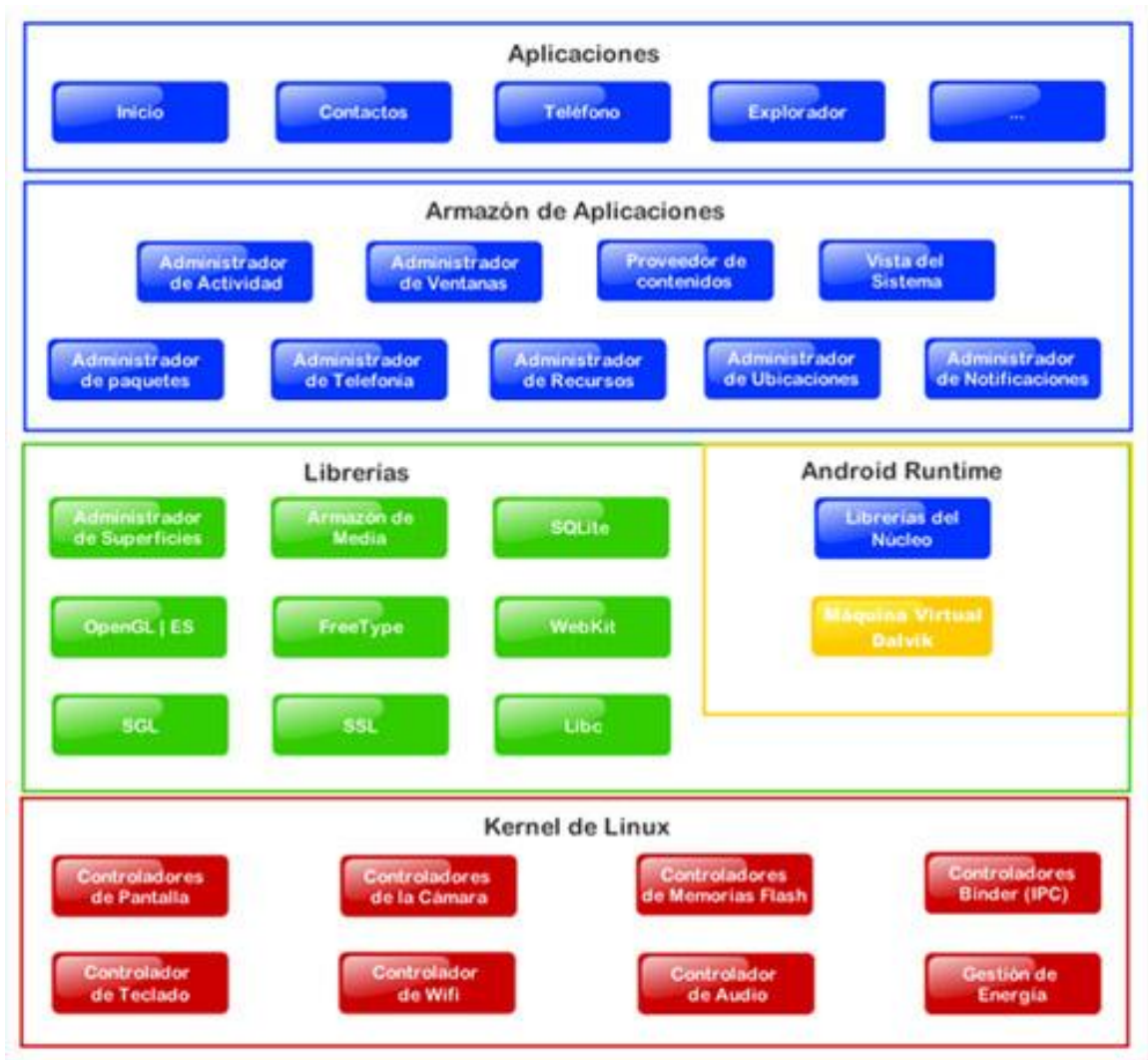


Ilustración 3: Arquitectura de Android

## ***Aplicaciones***

La capa superior de la *Ilustración 2* correspondiente a las aplicaciones, incluye todas las aplicaciones que contiene el dispositivo. Estas aplicaciones son todas las que contiene ya sean tienen interfaz de usuario o las que no la contienen que son las nativas programadas en lenguajes como C o C++, las administradas que son programadas en java que vienen de serie en el dispositivo o las que ha instalado el usuario. Entre estas aplicaciones se encuentran las principales del sistema como son los *launchers* o los *widgets*.

En esta capa de la arquitectura podemos ver como se alojan las aplicaciones ya sean las nativas de Android, que proporciona Google, así como las que incluye de serie el fabricante del teléfono o las que instala después el usuario utilizan el mismo marco de aplicación para acceder a los servicios que proporciona el sistema operativo. Esto sirve para que podamos crear aplicaciones que usen los mismos recursos que usan las aplicaciones nativas y así poder remplazar cualquiera de las aplicaciones del teléfono por otra de nuestra elección que nos guste más., siendo esta la gran diferencia de Android con respecto al resto de competencias ya que el control de las aplicaciones en su totalidad recae sobre el usuario.

## ***Marco de aplicaciones***

En la siguiente capa que se aprecia en la *Ilustración 2*, se pueden ver todas las clases y servicios que utilizan directamente las aplicaciones que se ejecutan en el dispositivo con el objetivo de poder realizar sus funciones y que se apoyan de la siguiente capa que como se mostrará son las bibliotecas y en el entorno de ejecución. La mayoría de los componentes de esta capa son bibliotecas del propio API de Java que acceden a los recursos a través de la máquina virtual. La arquitectura está pensada para que se puedan reutilizar los diferentes componentes de forma rápida y sencilla. Esto ha sido diseñado de tal forma que no se generen cientos de componentes de aplicaciones distintas, que respondan a la misma acción, dando la posibilidad de a los programas que sean modificados o remplazados por cualquier usuario, servicio o aplicación que requiera de su uso.

## ***Bibliotecas***

En esta se va a situar al igual que se vio en la imagen anterior justo sobre el kernel. Es la capa que contiene distintas librerías y bibliotecas nativas de Android. Estas bibliotecas están escritas en C o C++ y compiladas para la arquitectura hardware específico del teléfono, ya que la arquitectura de cada teléfono es distinta para cada modelo, así que por lo tanto, el fabricante es el que se encarga de realizar esta tarea. La característica principal de esta capa es la de proporcionar funcionalidad a las aplicaciones, para tareas que se repiten con frecuencia, evitando tener que codificarlas cada vez y garantizando que se llevan a cabo de la forma más eficiente

## ***Kernel de Linux***

Esta capa, es la que contiene el kernel Linux modificado versión 2.6, similar al que puede incluir cualquier distribución de Linux, como Ubuntu, Debian, Red Hat... con la diferencia que está adaptada al hardware en el que se va a ejecutar que en este caso es la de un teléfono inteligente bajo el sistema operativo Android. Esta capa proporciona una abstracción para los elementos hardware a los que tienen que acceder las aplicaciones, permitiendo acceder a esos componentes sin necesidad de conocer el modelo o características precisas de los que están instalados en cada teléfono. Para cada elemento hardware del teléfono existe un controlador dentro del kernel que permite utilizarlo desde el software. Además de proporcionar controladores hardware, el kernel se encarga de gestionar los diferentes recursos del teléfono y del sistema operativo.

## **2.3. SOAP**

SOAP se puede ver como un protocolo que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Fue creado por Microsoft, IBM y otros y está actualmente bajo el auspicio de la W3C. Es uno de los protocolos utilizados en servicios Web.

### **2.3.1. Introducción a SOAP**

Básicamente SOAP puede formar la capa base de una "pila de protocolo de web service", ofreciendo un framework de mensajería básica en la cual los Web Services se puedan construir. Este protocolo basado en XML consiste de tres partes:

- Un sobre o envoltorio, el cual define que hay en el mensaje y como procesarlo.
- Un conjunto de reglas de codificación para expresar instancias de tipos de datos.
- Una conversión para representar llamadas a procedimientos y respuestas.

A su vez el protocolo SOAP tiene tres características fundamentales:

- **Extensibilidad:** seguridad y WS-routing son extensiones aplicadas en el desarrollo).
- **Neutralidad:** SOAP puede ser utilizado sobre cualquier protocolo de transporte como HTTP, SMTP, TCP o JMS.
- **Independencia:** SOAP permite cualquier modelo de programación.

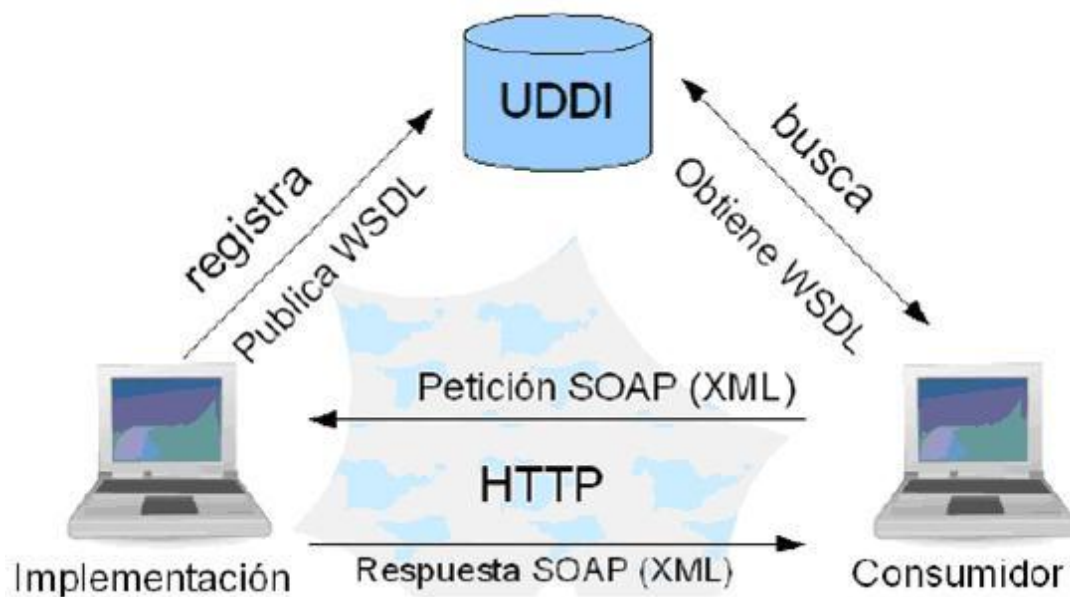


Ilustración 4: Arquitectura de SOAP

Como ejemplo de como los procedimientos SOAP pueden ser utilizados, un mensaje SOAP podría ser enviado a un sitio web que tiene habilitados web service, para realizar la búsqueda de algún precio en una base de datos, indicando los parámetros necesitados en la consulta. El sitio podría retornar un documento formateado en XML con el resultado, ejemplo, precios, localización, características. Teniendo los datos de respuesta en un formato estandarizado parseable, este puede ser integrado directamente en un sitio web o aplicación externa. La arquitectura SOAP consiste de muchas capas de especificación: para el formato del mensaje, *MEP (Message Exchange Patterns)*, subyacentes enlaces de protocolo de transporte, modelo de procesamiento de mensajes, y extensibilidad del protocolo. SOAP es el sucesor de *XML-RPC*, a pesar de que toma el transporte y la neutralidad de la interacción y el *envelope / header / body* de otra parte (probablemente de *WDDX*).

### 2.3.2. Historia de SOAP

Fue designado como un protocolo de acceso a objetos en 1998 por Microsoft. La especificación SOAP actualmente es mantenida por el *XML Protocol Working Group* del *World Wide Web Consortium*. Significa "*Simple Object Access Protocol*". El acrónimo se confunde a veces con SOA, siglas de arquitectura orientada a servicios, pero las siglas no están relacionadas. Después que SOAP se introdujo por primera vez, se convirtió en la capa subyacente de un conjunto más complejo de los web services, basada en la *WSDL (Web Services Description Language)* y *UDDI (Universal Description Discovery and Integration)*. Estos servicios, especialmente *UDDI*, han demostrado ser de mucho menos interés, pero una apreciación de ellos da una comprensión mas completa del esperado rol de SOAP comparado a como los web services están actualmente desarrollados.

### 2.3.3. Selección del protocolo

En este apartado se van a comentar los motivos de haber elegido SOAP frente a otros protocolos a la hora de usar las comunicaciones como son JSON, explicando y justificando una serie de decisiones que se han tomado sobre las distintas plataformas disponibles en el mercado de los servidores web de forma que se ajusten a las necesidades que se tenían en torno al Software y Hardware del proyecto.

Antes de empezar la comparación hay que dejar claro que es JSON. Esta basado en el acrónimo de *JavaScript Object Notation*. Es un formato alternativo de envío y recepción de datos, es decir remplaza a XML enviando en su lugar texto plano.

- Existe una librería en Android llamada *ksoap2* la cual hace más fácil las comunicaciones entre las aplicaciones cliente y el servicio web, por lo que en este caso las comunicaciones a través de SOAP se hacen mucho más sencillas que en JSON. En su lugar para JSON existe para Android una librería llamada *Gson*.
- SOAP está más preparado para ser usado con tecnologías tales como .NET o Java ya que los Servicios Web son mucho más sencillos de generar que utilizando JSON.
- SOAP genera los resultados en XML que contiene una definición específica del tipo de dato que hace que sea un protocolo más estricto.
- JSON una vez creado, es más fácil realizar cambios en el lado del cliente o el servidor, que SOAP debido a que SOAP necesita de múltiples cambios de código en ambos lados por cada cambio que se quiera realizar.
- A veces SOAP puede tener respuestas demasiado complejas y difíciles de interpretar al contrario de JSON que al ser en formato de texto plano, sus respuestas son más sencillas y fáciles de tratar.
- La seguridad de JSON es más crítica que la de SOAP, siendo un requisito necesario en este proyecto.
- El formato de las respuestas no está estandarizado por lo que no hay tipos de datos y a la hora de interpretar los resultados puede ser un problema a la hora de realizar la implementación.
- JSON a diferencia de SOAP es muy ligero conteniendo en sus respuestas, el contenido necesario que se tiene que enviar, no como SOAP que al ser en formato XML envía mucho metadatos de forma innecesaria.

Al analizar las diferencias anteriores y teniendo en cuenta que se quiere realizar un Servicio Web utilizando .NET con Visual Studio, el uso de SOAP se hace mucho interesante siempre que se tenga claro que datos se quieren usar y que estos no sean



excesivamente complejos. Como el formato de las respuestas no será demasiado complejo, el uso de SOAP hace que sea ideal a la hora de ser usado en el proyecto que se describe en este proyecto. Por lo tanto a la hora de desarrollar el proyecto fin de carrera se va a usar el protocolo SOAP visto en este apartado, a la hora de implementar las comunicaciones entre los clientes y el servidor.

## 2.4. .Net

.NET es un framework creado por la compañía Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones. Se compone en dos elementos principales: *Common Language Runtime* y un conjunto unificado de bibliotecas de clases, tales como ASP.NET para aplicaciones Web y servicios *Web XML*, *Windows Forms* para aplicaciones cliente inteligentes y *ADO.NET* para el acceso a datos sin rigidez.

### 2.4.1. Introducción a .Net

Para entender que es .NET, hay que comentar que es un conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando durante los últimos años con el objetivo de obtener una plataforma eficiente y eficaz que sea capaz de distribuir y unificar el software en forma de servicios y que puedan comunicarse y combinarse unos con otros de manera totalmente independiente de la plataforma, lenguaje de programación y modelo de componentes con los que hayan sido desarrollados.

.NET puede entenderse como un componente software que puede ser añadido al sistema operativo Windows. Contiene un extenso conjunto de soluciones predefinidas para necesidades generales de la programación de aplicaciones, y administra la ejecución de los programas escritos específicamente con la plataforma. Esta solución es el producto principal en la oferta de Microsoft, y pretende ser utilizada por la mayoría de las aplicaciones creadas para la plataforma Windows.

Las características que definen al framework de Microsoft son:

- Ofrece una nueva manera de desarrollar aplicaciones de sobremesa usando las clases Windows Forms.
- Ofrece una nueva manera de desarrollar aplicaciones basadas en navegador Web a través de ASP.NET.
- Las clases ADO.NET proveen una arquitectura desconectada para acceso a datos a través de Internet.
- Soporta la creación de Servicios Web XML independientes de la plataforma, a través de SOAP y WSDL.
- Además ofrece una nueva arquitectura para el desarrollo y explotación de objetos remotos.

.NET es considerado como la respuesta de Microsoft a lenguaje de programación Java, aunque tiene bastantes diferencias. Reúne en una misma plataforma un conjunto interesante de características, como independencia de plataforma, independencia de lenguaje, soporte de bases de datos, soporte para XML, Servicios Web y aplicaciones Web, entre otras. De entre todas estas, quizás una de las más importantes sean las dos primeras, referentes a la independencia tanto del lenguaje como de la plataforma.



**Ilustración 5: Microsoft .Net**

Con NET se puede desarrollar aplicaciones en múltiples lenguajes dentro de la plataforma pero lo más interesante, es que una aplicación puede tener diferentes partes desarrolladas en diferentes lenguajes de dentro del propio framework de Microsoft, y todas estas pueden comunicarse entre sí, transparentemente, sin tener que utilizar ningún tipo de capa intermedia que posibilite esta comunicación. Esto permite a su vez una gran reutilización de código, ya que las clases desarrolladas para un proyecto en un lenguaje concreto, podrán ser reutilizadas en un nuevo proyecto, independientemente del lenguaje en el que se desarrolle éste.

Al igual que en Java, el código .NET no se compila a código máquina, sino a un código en un formato intermedio, independiente de la plataforma. Esto permite llevar los binarios producidos de una plataforma a otra, tal como sucede en Java. Para que esto sea posible, así como en Java existe la máquina virtual (*JVM*), que es la encargada de interpretar o compilar JIT los bytecodes de una aplicación, en la plataforma .NET existe el *Common Language Runtime (CLR)*, que se encarga de ejecutar el código intermedio o *Common Intermediate Language (CIL)*.



### 2.4.2. Historia de .NET

.NET desde que salió al mercado ha ido mejorando versión a versión el framework de aplicaciones que sacó para desarrollar, de forma que las nuevas tecnologías que va incorporando o mejorando año tras año, hacen que se vayan cubriendo las necesidades que se requieren. Por este motivo, se ha encontrado la necesidad de explicar en este apartado de forma breve y concisa, la historia de .NET, centrándome en las diferentes versiones que han existido desde que fue lanzado, viendo las mejoras que ha habido versión tras versión.

Versión	Descripción
<b>.NET Framework 1.0</b>	Liberado el 13 de Febrero de 2002 para Win98, Me, NT 4.0, 2000 y XP.
<b>.NET Framework 1.1</b>	Publicada en Abril de 2003. Se mejoro: <ul style="list-style-type: none"><li>• Los controles ASP.NET que en el 1.0 eran un Addon ahora vienen nativos.</li><li>• Seguridad mejorada con Code Access Security para ASP.NET</li><li>• Soporte nativo para ODBC y Oracle</li><li>• Nacimiento del .NET Compact Framework</li><li>• Soporte a IPV6</li><li>• Cambios numerosos en las APIs.</li></ul>
<b>.NET Framework 2.0</b>	Liberado con Visual Studio 2005. Se mejoro: <ul style="list-style-type: none"><li>• Los fabulosos Generics</li><li>• Soporte para compilaciones de 64 bits</li><li>• Numerosos cambios en las APIs</li><li>• Integración con SQL Server: CLR Integration</li><li>• Inclusión del Runtime de .NET en aplicaciones nativas.</li><li>• Nuevos y mejorados controles de ASP.NET</li><li>• Controles de Datos con Data Binding automático.</li><li>• Soporte para Temas, Skins y Webparts.</li><li>• Nacimiento del .NET Micro Framework.</li><li>• Clases Parciales!</li><li>• Tipos “Nullable” (por ejemplo enteros a los que se les puede dar el valor de nulo)</li><li>• Métodos Anónimos!</li><li>• Iteradores</li><li>• Tablas de Datos</li></ul>
<b>.NET Framework 3.0</b>	Fue liberado en noviembre de 2006 e incluyó un nuevo conjunto de APIs hechas en código administrado que

	son parte integral de Windows Vista y Windows Server 2008.
<b>.NET Framework 3.5</b>	Fue liberado en noviembre de 2007. Modifica el CLR fundamental sobre el que se venía trabajando (2.0) para agregarle los métodos y propiedades requeridas sobre todo para LINQ. Aquí sí se liberó el Compact Framework 3.5. El código fuente de esta versión fue parcialmente liberado para conocimiento público con fines de depuración.
<b>.NET Framework 4.0</b>	Anunciado en Septiembre de 2008, con beta liberada el 20 de mayo de 2009 y finalmente en producción el 12 de Abril de 2010 es la versión actual del Framework y entre otras mejoras incluye: <ul style="list-style-type: none"> <li>• Extensiones de Paralelismo: Mejoran la computación paralela orientada a procesadores Multi-Core o sistemas distribuidos.</li> <li>• Parallel LINQ (PLINQ)</li> <li>• Task Parallel Library</li> </ul>

Tabla 2: Versiones de Android

### 2.4.3. Arquitectura de .Net

La arquitectura se divide en 5 aspectos tal y como puede verse en la figura siguiente. Dichos componentes serán comentados a continuación.

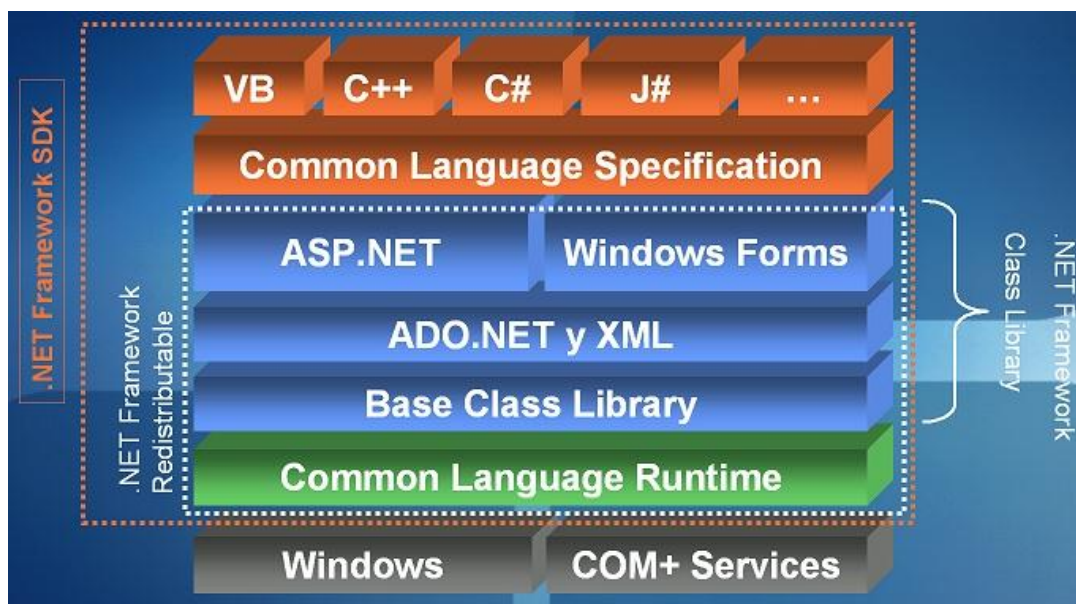


Ilustración 6: Arquitectura de .NET

Sus características son las que se muestran a continuación:

- **Entorno común de ejecución (Common Language Runtime, CLR):** Administra el código en tiempo de ejecución y proporciona los servicios básicos.
- **Biblioteca de clases base (Base Class Library, BCL):** Colección de código OO que puede ser empleado desde cualquier lenguaje .NET. Contiene los tipos básicos, clases para la entrada/salida, seguridad. Al tener definidos los tipos de datos para todos los lenguajes, facilita el intercambio de datos entre aplicaciones desarrolladas en distintos lenguajes.
- **Capa de datos y XML:** Gestiona el acceso a los datos y el tratamiento de datos XML. Los datos los gestiona mediante ADO.NET. Gran parte de la información de NET (configuración parte de la información de .NET (configuración, estructura de archivos y de aplicaciones) se gestiona mediante XML.
- **Facilidad para importar, exportar y tratar datos de/hacia XML.**
- **Definición de la interfaz: ASP.NET y Windows Forms:** ASP, ASP.NET utiliza Web Forms (para aplicaciones Web .NET utiliza Web Forms (para aplicaciones Web basadas en ASP) y los servicios Web. Windows Forms proporciona un conjunto de componentes de interfaz para desarrollar aplicaciones cliente basadas en el sistema operativo Windows.

#### 2.4.4. Selección de la plataforma

En este apartado se van a comentar los motivos de haber elegido .NET frente a otros IDE's como son NetBeans. Esto va a ser explicado y se justificará mediante una serie de decisiones que se han tomado sobre las distintas plataformas disponibles en el mercado de los servidores web de forma que se ajusten a las necesidades que se tenían en torno al Software y Hardware del proyecto.

- El motivo principal es que .NET funciona de forma excelente con el servidor IIS, mientras que NetBeans sería más apropiado para ser usado con Tomcat.
- Como se quiere crear un Servicio Web, este puede ser creado de una forma muy fácil a través de las herramientas que este IDE contiene, mientras que NetBeans suele ser usado para aplicaciones de Escritorio.
- .NET ocupa mucho más espacio que un proyecto en NetBeans.

Por estos motivos se va a utilizar .NET, ya que se integra mucho mejor con el servidor IIS, aparte de contener herramientas mucho más eficaces para crear Servicios Web ya que es uno de los objetivos que se buscan en este proyecto.

## 2.5. Internet Information Server

Internet Information Services (IIS) es un servidor Web desarrollado por la compañía Microsoft que ofrece una infraestructura de gran fiabilidad, capacidad de manejo y escalabilidad para aplicaciones Web sobre todas las versiones de su sistema operativo Windows. Además ofrece todos los servicios de servidores web como el que se persigue en este proyecto de fin de carrera.

### 2.5.1. Introducción

En sus inicios IIS era parte del *Option Pack* de Windows NT pero finalmente por su uso fue integrado en los sistemas operativos de Microsoft haciendo que la comunidad de usuarios que utilizan los distintos sistemas operativos de Microsoft, hagan uso de una forma fácil y cómoda de los servidores IIS que se ofrecen. Este servicio convierte a un ordenador de uso común, en un servidor web para Internet, haciendo que en los ordenadores que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente como es el caso de estudio que interesa en este proyecto fin de carrera.



Ilustración 7: Internet Information Server

Entre los servicios que se pueden encontrar en Internet Information Services, se proporcionan las herramientas y funciones necesarias para administrar de forma sencilla un servidor web seguro. Está basado en varios módulos ofreciendo gran capacidad para procesar distintos tipos de páginas, como es el caso de las páginas basadas en la tecnología ASP y ASP.NET. Además se pueden crear otro tipo de páginas con lenguajes libres como son el caso de PHP y Perl. Entre los servicios que se ofrecen al instalar en una máquina IIS están, la creación de servidores FTP,SMTP, NNTP y HTTP/HTTPS. En el caso de este proyecto, tal y como se va a ver posteriormente, se va a hacer uso principalmente de un servidor HTTP.

### 2.5.2. Historia

En este apartado de forma breve se describirá la historia del servidor creado por Microsoft:

Versión	Descripción
<b>IIS 1.0</b>	Internet Information Server fue liberado como un conjunto de servicios basados en Internet para Windows NT 3.51.
<b>IIS 2.0</b>	En esta versión se agregó un soporte para el sistema operativo Windows NT 4.0.
<b>IIS 3.0</b>	Introdujo las Active Server Pages que es una tecnología de scripting dinámico.
<b>IIS 4.0</b>	En esta versión se eliminó el soporte para el protocolo Gopher y fue puesto con Windows NT como un CD-ROM de "Paquete Opcional" por separado.
<b>IIS 5.0</b>	Se integró dentro de Windows XP Professional. Es una versión compacta del IIS que soporta sólo 10 conexiones simultáneas y sólo un sitio web.
<b>IIS 6.0</b>	Contiene soporte para IPv6.
<b>IIS 7.0</b>	Se añadió soporte para Windows Vista y Windows 7, que viene de forma preinstalada. No limita el número de conexiones permitidas pero si el del flujo de tareas basándose en las solicitudes activas concurrentes. Se mejora el uso y el rendimiento en escenarios P2P.

Tabla 3: Versiones de IIS

### 2.5.3. Selección de la plataforma

En este apartado se van a comentar los motivos de haber elegido IIS frente a otros servidores, explicando y justificando una serie de decisiones que se han tomado sobre las distintas plataformas disponibles en el mercado de los servidores web de forma que se ajusten a las necesidades que se tenían en torno al Software y Hardware del proyecto.

En todo momento se pensó en dos tipos de servidores, Apache e IIS, para el desarrollo del proyecto debido a su sencillez, madurez e integración en los mercados. Ambos servidores se integran bastante bien con todo tipo de comunicaciones y bases de datos y funcionan de forma excelente con aplicaciones Android como es el caso de estudio, por lo que cualquiera de las dos inicialmente sirve para el desarrollo del sistema e integrarlo así con el servicio web y la base de datos.



**Ilustración 8: IIS vs Apache**

Por estas razones se a continuación se van a mostrar las diferencias entre un servidor Apache y un servidor IIS, para que finalmente se pueda tomar una decisión.

- Los servidores IIS están basados en sistemas Windows, los cuales ya vienen preinstalados con dichos servidores haciendo que su instalación y configuración sea mucho más sencilla debido a que se adaptan muy bien con este tipo de sistemas operativos. En cambio los servidores Apache, son multiplataforma sirviendo para la gran familia de sistemas operativos basados en Unix así como para los principales sistemas operativos basados en Windows.
- El servidor IIS se integra muy bien con tecnologías potentes y modernas como son .NET haciendo el trabajo más sencillo al usuario, mientras que los servidores Apache utilizan tecnologías que son potentes como PHP o Perl pero que cuesta más integrarlas después de instalar y configurar el servidor.
- El soporte y el mantenimiento de cada uno de los dos servidores es muy distinto, siendo el servidor IIS soportado por personal responsable de la compañía Microsoft mientras que en Apache el soporte lo da la comunidad, siendo más difícil su depuración y gastando un mayor tiempo.

- Apache utiliza hosts virtuales para cada sitio web, mientras que en IIS esto no ocurre así sino que utiliza encabezados HTTP. Como no se quiere crear un host virtual, sino que con utilizar encabezados HTTP es suficiente para asegurar los datos a través de internet entre el cliente y el servidor, el protocolo de comunicaciones que tiene el servidor IIS es suficiente.
- Los servidores IIS aceptan nativamente ASP, en cambio Apache no acepta este lenguaje sino que utiliza otros como PHP o Perl. El uso de ASP es una ventaja debido a que en un futuro, si se quisiera implementar una página web que esté en contacto con el servicio web, así como con el servidor, no habría ningún problema ya que es muy fácil de usar y se integra perfectamente con los sistemas Windows e IIS.
- En cuanto a las bases de datos Apache acepta un gran número de bases de datos como es el caso de MySQL u Oracle. En el caso de IIS también acepta gran número de bases de datos al igual que Apache como son las bases de datos SQL EXPRESS

Por estos motivos se cree que por su simplicidad a la hora de instalarlo, configurarlo y administrarlo, es mucho más conveniente el uso de un servidor IIS frente a otros como son Apache, aunque este tipo de servidores servirían igualmente para crear el sistema debido a su gran número de ventajas.

## **2.6. SQL Express**

Microsoft SQL Express es un sistema de administración de bases de datos que ofrece un variado conjunto de características, protección de datos y rendimiento para clientes de aplicaciones de escritorio, aplicaciones web y almacenes de datos locales.

### **2.6.1. Introducción a SQL Express**

Microsoft SQL Express está diseñado para una implementación sencilla tal y como es el objeto de este proyecto y una creación de prototipos rápida de forma que al usuario le resulte relativamente fácil su instalación y configuración su mantenimiento y gestión, A la hora de tener un servidor IIS y usar el Framework .NET, esta base de datos se encuentra disponible de forma gratuita. Esta base de datos incluye un potente entorno gráfico para su administración, que permite el uso de comandos LDD y LMD gráficamente. Además permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y los terminales o clientes de la red sólo acceden a la información.



### 2.6.2. Selección de la base de datos

En este apartado se explican los motivos y decisiones justificadas a la hora de elegir y usar una base de datos. Las dos bases de datos que se han elegido inicialmente son, SQL Express y MySQL.

- En primer lugar decir que al elegir como servidor IIS tal y como se vio en apartados anteriores, la base de datos SQL Express se integra mucho mejor debido a que simplemente hace falta instalar el Framework de .NET para poder hacer uso de la base de datos. En cambio la configuración requerida para hacer uso de MySQL es mayor, así como la integración con .NET.
- Al usar como lenguaje de programación C#, esta se va a adaptar mejor con bases de datos del estilo de SQL Express, ya que MySQL se adapta a otros lenguajes como son Pearl. Con este punto decimos que la herramienta hace mucho a la hora de elegir la base de datos final.
- MySQL es más ligero, ocupa menos espacio a la hora de guardar los datos que SQL Express además de ser ligeramente más rápido.
- MySQL no soporta disparadores mientras que SQL Express si lo hace.

Como se ha podido ver, se elige como base de datos SQL Express debido a que se integra mucho mejor con el servidor elegido, así como el lenguaje que se usará como conector en el servicio web.

Otro motivo para su elección es que se va a utilizar como servidor de correo y IIS es ideal para ello. Al combinarlo con SQL Express se pueden obtener muy fácilmente los datos haciendo las consultas de forma óptima.





## 3. Análisis del sistema



### **3.1. Introducción**

En este apartado se va a definir el análisis del sistema a desarrollar, basándose dicho documento en los estándares que se han recogido de la Agencia Espacial Europea en su versión Lite, para los distintos proyectos de Ingeniería del Software. Se han modificado algunos apartados de la ESA debido a que se han evaluado y analizado, observando que eran innecesarios. Por ese motivo se ha decidido escoger la versión Lite de la ESA, que está especialmente diseñada para proyectos más pequeños como puede ser este que se desarrolla.

Una vez establecidas las tecnologías que se van a usar en el desarrollo del sistema, se van a definir de forma detallada cada una de las funcionalidades requeridas del sistema, de forma que se va a dar una descripción general del mismo para que el lector tenga una visión aproximada de cómo será el sistema final desarrollado.

En esta parte del documento se van a especificar las condiciones generales que tiene el proyecto, así como sus capacidades. Por último se detallarán todos y cada uno de los requisitos de usuario seguidos de los casos de uso, en los cuales se mostrarán los distintos usuario con sus posibles acciones al interaccionar con el sistema.

Por últimos se detallaran los requisitos software que se han recogido y analizado a partir de lo expuesto anteriormente.

### **3.2. Descripción General**

#### **3.2.1. Capacidades generales**

En este apartado se comenta cuál es el objetivo que debe de cumplir el sistema de cara a los usuarios. Por ello a continuación se muestran cuales son las capacidades que se han de cumplir de cara al análisis realizado inicialmente, que en sucesivos apartados se detallará mas profundamente.

- El sistema contiene dos tipos de usuarios, los administradores y los usuarios comunes que tienen menos privilegios, y por lo tanto son a los que se les van a prestar los libros.
- El sistema permite escanear en todo momento mediante la cámara del dispositivo, tanto a usuarios a través de su carnet de estudiante, como a los libros mediante su código de barras y etiquetas QR.
- El usuario administrador puede en todo momento gestionar libros y usuarios, de forma que es capaz de añadir, borrar, modificar y consultar cada uno de ellos.
- Se da la posibilidad de realizar un préstamo en todo momento, ya sea en la aplicación del administrador como en la de los usuarios.
- Únicamente el usuario administrador permite gestionar la devolución de libros.

- El usuario podrá acceder a la información de los préstamos que tiene, así como en caso de que tuviese penalizaciones, del tiempo que le queda para poder volver a realizar un préstamo.
- En todo momento se da la posibilidad de gestionar el cambio de contraseña para cada tipo de usuario final.
- El sistema proporciona mecanismos de seguridad a la hora de realizar el acceso a la aplicación mediante servicios web.
- Se gestiona la privacidad de la información de cada usuario a través de la información guardada en base de datos mediante el servicio web, a través del protocolo SOAP.
- El usuario podrá renovar préstamos de libros que tenga en ese momento, así como realizar una reserva de aquellos libros que quiera que se le presten.

### **3.2.2. Restricciones Generales**

En este apartado se van a comentar las restricciones que se van a dar inicialmente en el sistema, que son necesarias para diseñar el sistema de control de la gestión de la biblioteca.

- El usuario que es administrador tendrá mayores privilegios que el resto de usuarios, de forma que sea el único capaz de gestionar a tanto los usuarios, como a los libros de la biblioteca.
- Las aplicaciones del administrador y la del usuario final serán diferentes de forma que se mantenga la seguridad frente a la aplicación que gestiona todo el sistema que es la del administrador.
- En las aplicaciones Android, no se guarda ningún dato referente a los usuarios, a excepción del identificador del usuario que ha accedido a la aplicación. Esto es debido a que se pueda mantener la seguridad e integridad de los datos almacenados en base de datos. El motivo por el cuál se guarda el identificador del usuario en la aplicación del cliente, es por el mero hecho de no pedirle al usuario que introduzca constantemente su identificador.

### 3.2.3. Características de los usuarios

En el sistema hay dos tipos de usuarios que son los destinados al sistema. El primer tipo es el usuario administrador y el segundo es el destinado a realizar un uso común de la aplicación. Cada uno de ellos tiene un rol y unos privilegios distintos que hacen que el uso que cada usuario hace de la aplicación sea distinto.

- **Administrador:** este usuario es el encargado de gestionar y administrar la aplicación. Puede añadir o eliminar usuarios, así como libros de la biblioteca. De esta forma, el rol de este usuario permite gestionar todos los datos que componen el conjunto de información de la base de datos del sistema. Es el usuario con más privilegios capaz de acceder a la aplicación.
- **Cliente:** son los usuarios finales del sistema que mediante la aplicación destinada a este colectivo pueden realizar gestiones como desde ver la información de un libro que les interese, hasta poder realizar un autoprestamo del libro de forma que se ahorre el tener que cogerlo dándoselo al bibliotecario. A su vez es capaz de poder visualizar sus préstamos actuales así como de los últimos que ha realizado. Por último este usuario puede renovar los préstamos de los libros que tiene en su poder, así como reservar los que quiera que se le presten.

### 3.2.4. Entorno operacional

En este apartado se analizan las tecnologías del sistema en torno a los distintos usuarios definidos en el apartado anterior. Se deben distinguir entre las necesidades de cada uno de los roles anteriores:

ADMINISTRADOR
Teléfono móvil con sistema operativo Android y conexión a internet de forma inalámbrica, ya sea mediante Wifi o conexión 3G. Se requiere tener instalada la aplicación del administrador en el dispositivo desde el que se va a usar.

Tabla 4: Entorno operacional del administrador

CLIENTE
Teléfono móvil con sistema operativo Android y conexión a internet de forma inalámbrica, ya sea mediante Wifi o conexión 3G. Se requiere tener instalada la aplicación del cliente en el dispositivo desde el que se va a usar.

Tabla 5: Entorno operacional del cliente

### 3.3. Requisitos de usuario

El objetivo de este apartado es detallar los requisitos de usuario para poder diseñar en futuros apartados el sistema de la gestión del control de préstamos de una biblioteca. Estos requisitos se dividen en de capacidad y de restricción, los primeros indican qué debe ser capaz de realizar el sistema y los segundos especifican las restricciones que establecen el cómo se realizan las tareas que fueron indicadas en los primeros. El formato utilizado en la especificación es el siguiente:

Identificador	XX-UR-YY
Título	
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	
Fuente	
Descripción	

Tabla 6: Plantilla de requisito de usuario

- **Identificador:** es el código único que identifica a cada requisito.
- **Título:** es el nombre único del requisito.
- **Descripción:** detalla la especificación del requisito de manera precisa y simple.
- **Necesidad:** establece la importancia del requisito desde el punto de vista del cliente. Los valores pueden ser esencial, conveniente u opcional.
- **Prioridad:** establece la importancia del requisito en función del desarrollo del proyecto. Los valores pueden ser alta, media o baja.
- **Fuente:** indica la procedencia del requisito.

### 3.3.1. Requisitos de Capacidad

<b>Identificador</b>	<b>RUC-01</b>
<b>Título</b>	Escanear un libro
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario puede capturar códigos de barras de los libros a través de la cámara, pudiendo obtener la información del libro.	

Tabla 7: RUC-01

<b>Identificador</b>	<b>RUC-02</b>
<b>Título</b>	Escanear un usuario
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario puede capturar códigos de barras de los carnets de estudiante a través de la cámara, pudiendo obtener la información del usuario en cuestión.	

Tabla 8: RUC-02

<b>Identificador</b>	<b>RUC-03</b>
<b>Título</b>	Acceso a la aplicación
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario puede acceder a la aplicación introduciendo su ‘usuario’ y su ‘contraseña’ en los campos del formulario correspondiente, accediendo a la información de su cuenta.	

Tabla 9: RUC-03

<b>Identificador</b>	<b>RUC-04</b>
<b>Título</b>	Consulta de libro escaneando código de barras
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario puede consultar un libro mediante el escaneo de código de barras.	

Tabla 10: RUC-04

Identificador	RUC-05
Título	Consulta de usuario escaneando código de barras
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
Un usuario administrador puede consultar un libro mediante el escaneo del código de barras de su carnet de estudiante.	

Tabla 11: RUC-05

Identificador	RUC-06
Título	Consulta de libro introduciendo ISBN
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
Un usuario puede consultar un libro metiendo de forma manual, el ISBN del libro en un formulario.	

Tabla 12: RUC-06

Identificador	RUC-07
Título	Consulta de usuario introduciendo su identificador de usuario
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
Un usuario administrador puede consultar un usuario metiendo de forma manual el ISBN del libro en un formulario.	

Tabla 13: RUC-07

Identificador	RUC-08
Título	Añadir un usuario
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
Un usuario administrador es capaz de añadir un usuario introduciendo los datos del mismo en un formulario.	

Tabla 14: RUC-08



<b>Identificador</b>	<b>RUC-09</b>
<b>Título</b>	Añadir un libro
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario administrador es capaz de añadir un libro introduciendo los datos del mismo en un formulario.	

Tabla 15: RUC-09

<b>Identificador</b>	<b>RUC-10</b>
<b>Título</b>	Borrar un usuario escaneando carnet de estudiante
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario administrador es capaz de eliminar un usuario de la base de datos escaneando el código de barras de su carnet de estudiante.	

Tabla 16: RUC-10

<b>Identificador</b>	<b>RUC-11</b>
<b>Título</b>	Borrar un usuario introduciendo el identificador de usuario
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario administrador es capaz de eliminar un usuario de la base de datos introduciendo en el formulario el identificador del usuario.	

Tabla 17: RUC-11

<b>Identificador</b>	<b>RUC-12</b>
<b>Título</b>	Borrar un libro escaneando mediante su código de barras
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario administrador es capaz de eliminar un libro de la base de datos escaneando su código de barras.	

Tabla 18: RUC-12

Identificador	RUC-13
Título	Borrar un libro introduciendo su ISBN
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
Un usuario administrador es capaz de eliminar un libro de la base de datos introduciendo en el formulario el.	

Tabla 19: RUC-13

Identificador	RUC-14
Título	Modificar un usuario escaneando el código de barras del carnet de estudiante
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
Un usuario administrador es capaz de modificar los datos de un libro escaneando el código de barras. Una vez escaneado se mostrará un formulario con los datos que se pueden modificar.	

Tabla 20: RUC-14

Identificador	RUC-15
Título	Modificar un usuario introduciendo su identificador de usuario
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
Un usuario administrador es capaz de modificar los datos de un libro introduciendo en un formulario el código identificador de usuario. Una vez validado el usuario se mostrará un formulario con los datos que se pueden modificar.	

Tabla 21: RUC-15

<b>Identificador</b>	<b>RUC-16</b>
<b>Título</b>	Modificar un libro escaneando el código de barras del libro
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario administrador es capaz de modificar los datos de un libro escaneando el código de barras. Una vez escaneado se mostrará un formulario con los datos que se pueden modificar.	

Tabla 22: RUC-16

<b>Identificador</b>	<b>RUC-17</b>
<b>Título</b>	Modificar un libro introduciendo su ISBN
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario administrador es capaz de modificar los datos de un libro introduciendo en un formulario el ISBN. Una vez validado el libro se mostrará un formulario con los datos que se pueden modificar.	

Tabla 23: RUC-17

<b>Identificador</b>	<b>RUC-18</b>
<b>Título</b>	Préstamo de un libro mediante escaneo de código de barras
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Cualquier tipo de usuario es capaz de generar un préstamo de un libro escaneando el código único del libro.	

Tabla 24: RUC-18

<b>Identificador</b>	<b>RUC-19</b>
<b>Título</b>	Préstamo de un libro mediante la introducción del ISBN
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Cualquier tipo de usuario es capaz de generar un préstamo de un libro introduciendo el ISBN del libro.	

Tabla 25: RUC-19

Identificador	RUC-20
Título	Devolución de un libro mediante escaneo de código de barras
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
El usuario administrador es capaz de gestionar la devolución de un préstamo de un libro escaneando el código único del libro.	

Tabla 26: RUC-20

Identificador	RUC-21
Título	Devolución de un libro mediante la introducción del ISBN
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
El usuario administrador es capaz de gestionar la devolución de un préstamo de un libro introduciendo el ISBN del libro.	

Tabla 27: RUC-21

Identificador	RUC-22
Título	Cambio de contraseña
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
Cualquier tipo de usuario es capaz de cambiar la contraseña de la aplicación. Para ello es necesario que se introduzca la contraseña actual y la contraseña nueva como medida de seguridad.	

Tabla 28: RUC-22

Identificador	RUC-23
Título	Visualizar historial de préstamos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Necesidad	Esencial
Fuente	Cliente
Descripción	
Los usuarios comunes son capaces de visualizar los últimos préstamos que han realizado	

Tabla 29: RUC-23

<b>Identificador</b>	<b>RUC-24</b>
<b>Título</b>	Visualizar préstamos actuales
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Los usuarios comunes son capaces de visualizar los préstamos que tienen actualmente, así como su fecha de devolución.	

Tabla 30: RUC-24

<b>Identificador</b>	<b>RUC-25</b>
<b>Título</b>	Visualización de información de usuario
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Los usuarios comunes son capaces de visualizar su información de usuario en la que se muestra a parte de sus datos, si tiene penalizaciones y préstamos vigentes.	

Tabla 31: RUC-25

<b>Identificador</b>	<b>RUC-26</b>
<b>Título</b>	Aviso de número máximo de préstamos
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
A la hora de realizar un préstamo, la aplicación avisará al usuario si este ya tiene el número máximo de préstamos que se le permite en su momento actual.	

Tabla 32: RUC-26

<b>Identificador</b>	<b>RUC-27</b>
<b>Título</b>	Aviso de penalización
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
A la hora de realizar un préstamo, si el usuario está penalizado, no se le permitirá realizar ningún préstamo hasta que finalice el periodo de penalización.	

Tabla 33: RUC-27

<b>Identificador</b>	<b>RUC-28</b>
<b>Título</b>	Aviso de préstamo nuevo por correo electrónico
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
A la hora de realizar un nuevo préstamo, se avisará al usuario por el correo electrónico de la universidad que ha realizado un préstamo.	

Tabla 34: RUC-28

<b>Identificador</b>	<b>RUC-29</b>
<b>Título</b>	Aviso de vencimiento de préstamo
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
El día anterior del vencimiento del préstamo, se avisará al usuario por el correo electrónico de la universidad, que su préstamo se finaliza.	

Tabla 35: RUC-29

<b>Identificador</b>	<b>RUC-30</b>
<b>Título</b>	Distintas aplicaciones para diferentes roles de usuario
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Dependiendo del rol de usuario, se accederá a diferentes aplicaciones: La del usuario o la del bibliotecario.	

Tabla 36: RUC-30

<b>Identificador</b>	<b>RUC-31</b>
<b>Título</b>	Distintos usuarios de la biblioteca.
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Habrá dos tipos de usuarios en la biblioteca que tendrán permisos de préstamo de libros diferentes:	
<ul style="list-style-type: none"><li>• Alumno</li><li>• Profesor</li></ul>	

Tabla 37: RUC-31

<b>Identificador</b>	<b>RUC-32</b>
<b>Título</b>	Periodo de préstamo de libros variable.
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
El periodo de préstamo de cada libro será variable dependiendo del tipo de usuario.	

Tabla 38: RUC-32

<b>Identificador</b>	<b>RUC-33</b>
<b>Título</b>	El número de libro máximo por usuario es variable.
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario podrá coger simultáneamente un número variable de libros dependiendo del tipo de usuario.	

Tabla 39: RUC-33

<b>Identificador</b>	<b>RUC-34</b>
<b>Título</b>	Renovar un préstamo.
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario podrá renovar un préstamo siempre que esté a tres días de su finalización.	

Tabla 40: RUC-34

<b>Identificador</b>	<b>RUC-35</b>
<b>Título</b>	Reservar un libro.
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario podrá reservar un libro siempre que no este prestado.	

Tabla 41: RUC-35

### 3.3.2. Requisitos de Restricción

<b>Identificador</b>	<b>RUR-01</b>
<b>Título</b>	Uso del sistema por usuarios registrados
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Únicamente las personas registradas pueden hacer uso del sistema.	

Tabla 42: RUR-01

<b>Identificador</b>	<b>RUR-02</b>
<b>Título</b>	Protección de datos personales
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
El sistema impedirá que se puedan acceder a la información confidencial de los usuarios registrados en el sistema	

Tabla 43: RUR-02

<b>Identificador</b>	<b>RUR-03</b>
<b>Título</b>	Idioma de la aplicación
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
El idioma del sistema será en castellano.	

Tabla 44: RUR-03

<b>Identificador</b>	<b>RUR-04</b>
<b>Título</b>	Tiempos de espera mínimos
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
El sistema debe responder en un tiempo breve a cada petición realizada desde el dispositivo móvil.	

Tabla 45: RUR-04



<b>Identificador</b>	<b>RUR-05</b>
<b>Título</b>	Funcionamiento en dispositivos Android
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Cada aplicación funcionará únicamente en dispositivos con el sistema operativo Android.	

Tabla 46: RUR-05

<b>Identificador</b>	<b>RUR-06</b>
<b>Título</b>	Funcionamiento a través de internet
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
El funcionamiento de la aplicación será a través de redes de internet inalámbricas.	

Tabla 47: RUR-06

<b>Identificador</b>	<b>RUR-07</b>
<b>Título</b>	Evitar registro automático
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Se impedirá que un sistema automatizado pueda crear cuentas de usuario.	

Tabla 48: RUR-07

<b>Identificador</b>	<b>RUR-08</b>
<b>Título</b>	Almacenar datos de inicio de sesión
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Los datos introducidos al inicio de sesión poder ser almacenados en el dispositivo con el fin de evitar al usuario introducirlos cada vez que se acceda.	

Tabla 49: RUR-08

<b>Identificador</b>	<b>RUR-09</b>
<b>Título</b>	Número máximo de préstamos
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Un usuario podrá realizar un máximo de cuatro préstamos con un máximo de tiempo de por ejemplar de dos semanas si es un profesor, o de dos libros en un periodo máximo de una semana por ejemplar si es un estudiante.	

Tabla 50: RUR-09

<b>Identificador</b>	<b>RUR-10</b>
<b>Título</b>	Penalización por entrega tarde de libros
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
Si un usuario se pasa del tiempo de préstamo de un libro, se aplicará una penalización de tiempo que empezará, desde el día que lo entregue hasta que se cumpla dicho plazo de penalización.	

Tabla 51: RUR-10

<b>Identificador</b>	<b>RUR-11</b>
<b>Título</b>	Etiquetas inteligentes a escanear
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
El sistema únicamente podrá escanear cierto tipo de etiquetas inteligentes.	

Tabla 52: RUR-11

<b>Identificador</b>	<b>RUR-12</b>
<b>Título</b>	Disponibilidad de Cámara de fotos
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
<b>Necesidad</b>	Esencial
<b>Fuente</b>	Cliente
<b>Descripción</b>	
El dispositivo móvil deberá de disponer de cámara de fotos.	

Tabla 53: RUR-12

### 3.4. Casos de uso

Los casos de uso son la representación gráfica de los requisitos, y por lo tanto, nacen de éstos. El objetivo de este apartado es representar y detallar las funcionalidades y tareas que puede realizar un usuario en el sistema, partiendo de la base de que el usuario controla el sistema mediante el dispositivo móvil que se ha especificado en los requisitos de usuario. Se trata de una técnica muy utilizada en el área de la Ingeniería del Software para mostrar la futura capacidad del sistema de una forma simple e intuitiva.

Cada caso de uso describe los pasos que debe seguir el actor (en este caso, el usuario) para lograr aplicar una determinada funcionalidad, además, también se detallan las condiciones necesarias y cómo se realiza la interacción aplicación móvil frente al servicio web.

#### 3.4.1. Diagramas de casos de uso

En este apartado se van a mostrar los diagramas que han sido necesarios establecer por cada tipo de usuario. Como se puede apreciar en cada diagrama, se representan a cada tipo de actor y las interacciones que estos hacen con la aplicación.

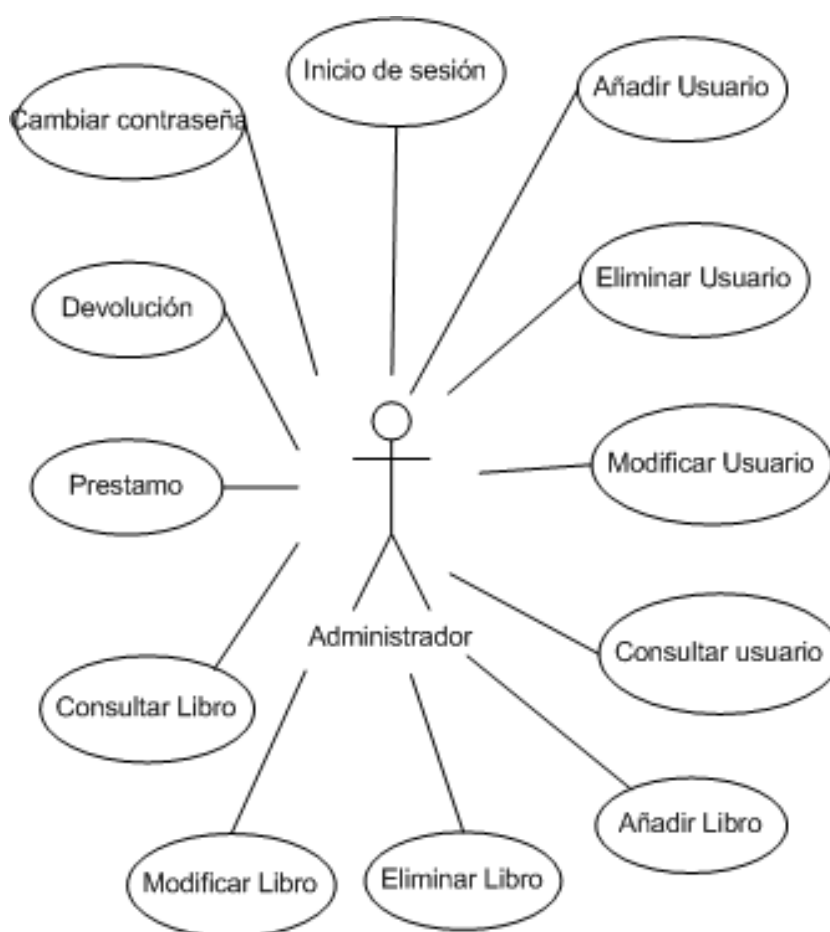


Ilustración 9: Diagrama de casos de uso - administrador

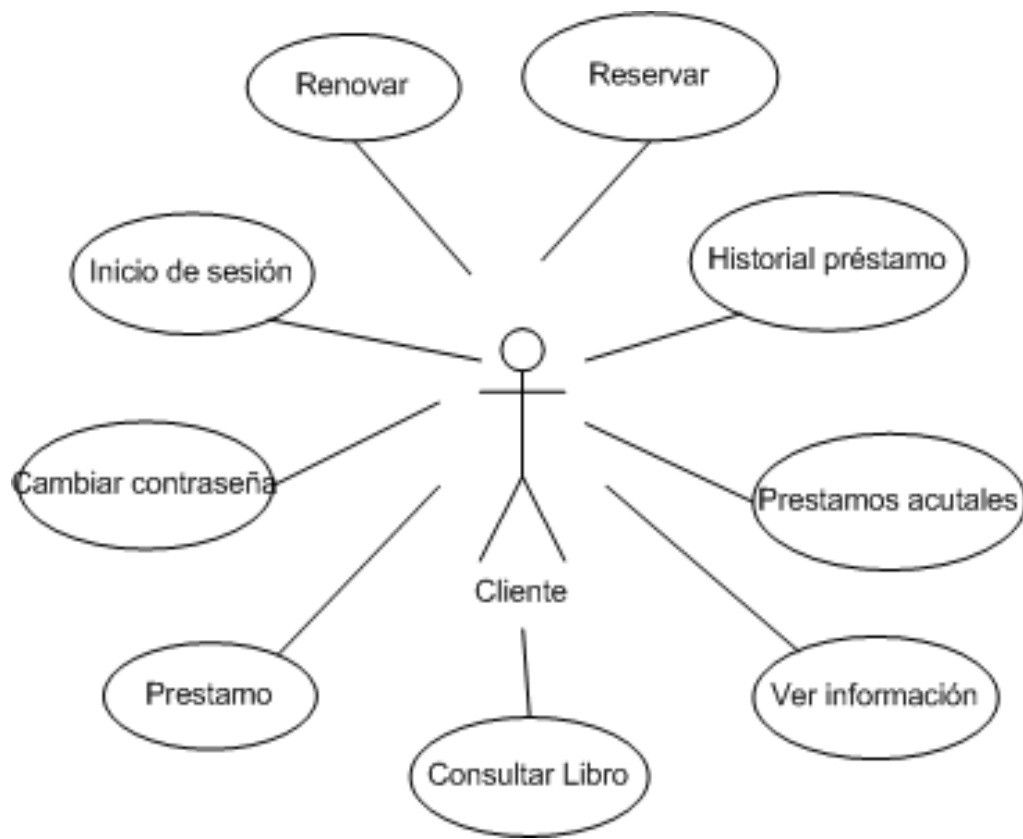


Ilustración 10: Diagrama de casos de uso – cliente

### 3.4.2. Descripción textual

En esta parte en la que describimos los casos de uso, la plantilla utilizada en la especificación de los casos de uso es la siguiente:

<b>Identificador:</b>	<b>CU-XX</b>
<b>Nombre:</b>	
<b>Actores:</b>	
<b>Objetivo:</b>	
<b>Descripción:</b>	
<b>Pre-condición:</b>	
<b>Post-condición:</b>	

Tabla 54: Plantilla de caso de uso

- **Identificador:** es el código único que identifica a cada caso de uso. Se compone de 2 partes:
  - **CU:** corresponde a '*Caso de Uso*'.
  - **XX:** número de caso de uso.
- **Nombre:** establece una identificación denominando el caso de uso en relación a su función primordial. Esta denominación se utilizará en los diagramas UML de casos de uso para la identificación.
- **Actores:** se refiere a los roles de personas o sistemas que podrán desempeñar una función en el sistema utilizando el caso de uso.
- **Objetivo:** establece la meta final del caso de uso.
- **Descripción:** especificación de los eventos correspondientes a la ejecución estándar del caso de uso.
- **Pre-condición:** especificación de condiciones necesarias para la ejecución del caso de uso.
- **Post-condición:** especificación de condiciones necesarias para la ejecución del caso de uso.

<b>Identificador:</b>	<b>CU-01</b>
<b>Nombre:</b>	Iniciar sesión
<b>Actores:</b>	Cliente y administrador
<b>Objetivo:</b>	Iniciar sesión dentro del sistema.
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario introduce su nombre de usuario y su contraseña.</li><li>2. El usuario pulsa el botón de Aceptar para iniciar la sesión.</li><li>3. Se envía la petición al servicio web que la valida y comprueba el resultado.</li><li>4. Si es correcto, se inicia sesión en el sistema.</li></ol>
<b>Pre-condición:</b>	Se accede a la aplicación desde el dispositivo móvil.
<b>Post-condición:</b>	Se accede a la aplicación correctamente

Tabla 55: CU-01

<b>Identificador:</b>	<b>CU-02</b>
<b>Nombre:</b>	Añadir usuario
<b>Actores:</b>	Administrador
<b>Objetivo:</b>	Añadir un usuario al sistema
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario administrador accede al sistema.</li><li>2. El usuario administrador pulsa sobre el botón de añadir un usuario.</li><li>3. El usuario administrador introduce los datos del usuario que se desea añadir</li><li>4. Se envía la petición al servicio web que la valida y añade el usuario a la base de datos.</li><li>5. El servicio web devuelve a la aplicación móvil que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	No existe el usuario que se desea registrar
<b>Post-condición:</b>	El usuario se ha añadido correctamente al sistema

Tabla 56: CU-02

<b>Identificador:</b>	<b>CU-03</b>
<b>Nombre:</b>	Eliminar usuario
<b>Actores:</b>	Administrador
<b>Objetivo:</b>	Eliminar un usuario del sistema
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario administrador accede al sistema.</li><li>2. El usuario administrador pulsa sobre el botón de eliminar un usuario.</li><li>3. El usuario administrador introduce el número identificativo del usuario que se desea eliminar</li><li>4. Se envía la petición al servicio web que la valida y elimina el usuario a la base de datos.</li><li>5. El servicio web devuelve a la aplicación móvil que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	El usuario que se desea eliminar existe previamente en el sistema
<b>Post-condición:</b>	El usuario se ha eliminado correctamente del sistema

Tabla 57: CU-03

<b>Identificador:</b>	<b>CU-04</b>
<b>Nombre:</b>	Modificar usuario
<b>Actores:</b>	Administrador
<b>Objetivo:</b>	Modificar un usuario del sistema
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario administrador accede al sistema.</li><li>2. El usuario administrador pulsa sobre el botón de modificar un usuario.</li><li>3. El usuario administrador introduce el número identificativo del usuario que se desea modificar</li><li>4. Se envía la petición al servicio web que la valida y devuelve los datos del usuario que es posible modificar.</li><li>5. El usuario administrador introduce los cambios del usuario que se desea modificar</li><li>6. Se envía la petición al servicio web que la valida y modifica el usuario a la base de datos.</li><li>7. El servicio web devuelve a la aplicación móvil que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	El usuario existe previamente en el sistema
<b>Post-condición:</b>	El usuario se ha modificado correctamente en el sistema

Tabla 58: CU-04

<b>Identificador:</b>	<b>CU-05</b>
<b>Nombre:</b>	Consultar usuario
<b>Actores:</b>	Administrador
<b>Objetivo:</b>	Visualizar un usuario en el dispositivo móvil
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario administrador accede al sistema.</li><li>2. El usuario administrador pulsa sobre el botón de consultar un usuario.</li><li>3. El usuario administrador introduce el número identificativo del usuario que se desea consultar</li><li>4. Se envía la petición al servicio web que la valida y devuelve los datos del usuario.</li><li>5. El servicio web devuelve a la aplicación móvil que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	El usuario existe previamente en el sistema
<b>Post-condición:</b>	Se ha visualizado al usuario en el dispositivo móvil

Tabla 59: CU-05

<b>Identificador:</b>	<b>CU-06</b>
<b>Nombre:</b>	Añadir libro
<b>Actores:</b>	Administrador
<b>Objetivo:</b>	Añadir un libro al sistema
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario administrador accede al sistema.</li><li>2. El usuario administrador pulsa sobre el botón de añadir un libro.</li><li>3. El usuario administrador introduce los datos del libro que se desea añadir</li><li>4. Se envía la petición al servicio web que la valida y añade el libro a la base de datos.</li><li>5. El servicio web devuelve a la aplicación móvil que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	No existe el libro que se desea registrar
<b>Post-condición:</b>	El libro se ha añadido correctamente al sistema

Tabla 60: CU-06



<b>Identificador:</b>	<b>CU-07</b>
<b>Nombre:</b>	Eliminar libro
<b>Actores:</b>	Administrador
<b>Objetivo:</b>	Eliminar un libro del sistema
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario administrador accede al sistema.</li><li>2. El usuario administrador pulsa sobre el botón de eliminar un libro.</li><li>3. El usuario administrador introduce el número identificativo del libro que se desea eliminar</li><li>4. Se envía la petición al servicio web que la valida y elimina el libro a la base de datos.</li><li>5. El servicio web devuelve a la aplicación móvil que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	El libro que se desea eliminar existe previamente en el sistema
<b>Post-condición:</b>	El libro se ha eliminado correctamente del sistema

Tabla 61: CU-07

<b>Identificador:</b>	<b>CU-08</b>
<b>Nombre:</b>	Modificar libro
<b>Actores:</b>	Administrador
<b>Objetivo:</b>	Modificar un libro del sistema
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario administrador accede al sistema.</li><li>2. El usuario administrador pulsa sobre el botón de modificar un libro.</li><li>3. El usuario administrador introduce el número identificativo del libro que se desea modificar</li><li>4. Se envía la petición al servicio web que la valida y devuelve los datos del usuario que es posible modificar.</li><li>5. El usuario administrador introduce los cambios del libro que se desea modificar</li><li>6. Se envía la petición al servicio web que la valida y modifica el libro a la base de datos.</li><li>7. El servicio web devuelve a la aplicación móvil que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	El libro existe previamente en el sistema
<b>Post-condición:</b>	El libro se ha modificado correctamente en el sistema

Tabla 62: CU-08

<b>Identificador:</b>	<b>CU-09</b>
<b>Nombre:</b>	Consultar libro
<b>Actores:</b>	Administrador y cliente
<b>Objetivo:</b>	Visualizar un libro en el dispositivo móvil
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario administrador accede al sistema.</li><li>2. El usuario administrador pulsa sobre el botón de consultar un libro.</li><li>3. El usuario administrador introduce el número identificativo del libro que se desea consultar</li><li>4. Se envía la petición al servicio web que la valida y devuelve los datos del usuario.</li><li>5. El servicio web devuelve a la aplicación móvil que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	El libro existe previamente en el sistema
<b>Post-condición:</b>	Se ha visualizado al libro en el dispositivo móvil

Tabla 63: CU-09

<b>Identificador:</b>	<b>CU-10</b>
<b>Nombre:</b>	Préstamo
<b>Actores:</b>	Administrador y cliente
<b>Objetivo:</b>	Préstamo de un libro a un usuario
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario accede al sistema.</li><li>2. El usuario pulsa sobre el botón de Préstamo.</li><li>3. El usuario introduce el número identificativo del libro que se desea prestar</li><li>4. Se envía la petición al servicio web que la valida y devuelve si es posible realizar el préstamo para dicho usuario.</li><li>5. En caso de que el préstamo sea posible, se devuelve a la aplicación, que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	El libro existe previamente en el sistema y no está prestado
<b>Post-condición:</b>	Se ha prestado el libro

Tabla 64: CU-10

<b>Identificador:</b>	<b>CU-11</b>
<b>Nombre:</b>	Devolución
<b>Actores:</b>	Administrador
<b>Objetivo:</b>	Devolución de un libro a un usuario
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario accede al sistema.</li><li>2. El usuario pulsa sobre el botón de Devolución.</li><li>3. El usuario introduce el número identificativo del libro que se desea devolver</li><li>4. Se envía la petición al servicio web que la valida y devuelve si es posible realizar la devolución para dicho usuario.</li><li>5. En caso de que la devolución sea posible, se devuelve a la aplicación, que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	El libro existe previamente en el sistema y está prestado
<b>Post-condición:</b>	Se ha devuelto el libro

Tabla 65: CU-11

<b>Identificador:</b>	<b>CU-12</b>
<b>Nombre:</b>	Cambiar contraseña
<b>Actores:</b>	Administrador y cliente
<b>Objetivo:</b>	Cambio de la contraseña del usuario en cuestión.
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario accede al sistema.</li><li>2. El usuario pulsa sobre el botón de Cambiar contraseña.</li><li>3. El usuario introduce su contraseña actual y la nueva por la que quiere realizar el cambio.</li><li>4. Se envía la petición al servicio web que la valida y devuelve si es posible realizar el cambio de contraseña.</li><li>5. En caso de que sea posible, se devuelve a la aplicación, que la operación ha sido correcta.</li></ol>
<b>Pre-condición:</b>	El usuario existe previamente en el sistema
<b>Post-condición:</b>	Se ha realizado el cambio de contraseña correctamente

Tabla 66: CU-12

<b>Identificador:</b>	<b>CU-13</b>
<b>Nombre:</b>	Historial préstamo
<b>Actores:</b>	Cliente
<b>Objetivo:</b>	Se muestra una lista con los últimos préstamos realizados
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario accede al sistema.</li><li>2. El usuario pulsa sobre el botón de Historial préstamo.</li><li>3. Se envía la petición al servicio web y si es correcta se devuelve al dispositivo móvil.</li><li>4. Se muestra una lista con los últimos libros prestados al usuario que se visualizan en el dispositivo móvil.</li></ol>
<b>Pre-condición:</b>	El usuario existe previamente en el sistema
<b>Post-condición:</b>	Se ha visualizado la lista de los préstamos.

Tabla 67: CU-13

<b>Identificador:</b>	<b>CU-14</b>
<b>Nombre:</b>	Préstamo actual
<b>Actores:</b>	Cliente
<b>Objetivo:</b>	Devolución de un libro a un usuario
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario accede al sistema.</li><li>2. El usuario pulsa sobre el botón de Préstamo actual.</li><li>3. Se envía la petición al servicio web y si es correcta se devuelve al dispositivo móvil.</li><li>4. Se muestra una lista con los libros prestados al usuario actualmente que se visualizan en el dispositivo móvil.</li></ol>
<b>Pre-condición:</b>	El usuario existe previamente en el sistema
<b>Post-condición:</b>	Se ha visualizado la lista de los préstamos.

Tabla 68: CU-14

<b>Identificador:</b>	<b>CU-15</b>
<b>Nombre:</b>	Ver información
<b>Actores:</b>	Cliente
<b>Objetivo:</b>	Cambio de la contraseña del usuario en cuestión.
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario accede al sistema.</li><li>2. El usuario pulsa sobre el botón de Ver información.</li><li>3. Se envía la petición al servicio web y si es correcta se devuelve al dispositivo móvil.</li><li>4. Se muestra la información del usuario.</li></ol>
<b>Pre-condición:</b>	El usuario existe previamente en el sistema
<b>Post-condición:</b>	Se ha visualizado la información del usuario

Tabla 69: CU-15

<b>Identificador:</b>	<b>CU-16</b>
<b>Nombre:</b>	Renovación
<b>Actores:</b>	Cliente
<b>Objetivo:</b>	Renovación de un préstamo
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario accede al sistema.</li><li>2. El usuario pulsa sobre el botón de Renovar eligiendo un libro.</li><li>3. Se envía la petición al servicio web y si es correcta se devuelve al dispositivo móvil.</li><li>4. Se muestra si ha sido posible la renovación del libro.</li></ol>
<b>Pre-condición:</b>	El usuario existe previamente en el sistema
<b>Post-condición:</b>	Se ha efectuado la renovación del libro.

Tabla 70: CU-16

<b>Identificador:</b>	<b>CU-17</b>
<b>Nombre:</b>	Reserva
<b>Actores:</b>	Cliente
<b>Objetivo:</b>	Reserva de un libro
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. El usuario accede al sistema.</li><li>2. El usuario pulsa sobre el botón de Reservar eligiendo el libro que se desea reservar.</li><li>3. Se envía la petición al servicio web y si es correcta se devuelve al dispositivo móvil.</li><li>4. Se muestra si ha sido posible la reserva del libro.</li></ol>
<b>Pre-condición:</b>	El usuario existe previamente en el sistema
<b>Post-condición:</b>	Se ha reservado el libro

Tabla 71: CU-17

### 3.5. Requisitos Software

En este apartado vamos a detallar los requisitos de software, tanto funcionales como no funcionales, obtenidos a través del análisis exhaustivo de los requisitos de usuario y de la información extraída por el cliente. Los tipos de requisitos que vamos a detallar son los siguientes:

- **Requisitos de software funcional:** Especifican el “qué” tiene que hacer el software, es decir, el propósito del software.
- **Requisitos de software no funcional:** Especifican el “cómo” tiene que hacerse y funcionar el software.

El formato seguido en la especificación de los requisitos es el siguiente:

RS(T)-XX			
Título			
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	
Necesidad	<input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
Estabilidad	<input type="checkbox"/> Sí <input type="checkbox"/> No	Verificable	<input type="checkbox"/> Sí <input type="checkbox"/> No
Descripción			

Tabla 72: Plantilla de los requisitos funcionales

- **RS(T) XX:** requisito de tipo (T) que puede ser funcional (F) o no funcional (NF). El número de requisito XX ha de ser único y no puede repetirse ningún código de la forma R(T) XX.
- **Título:** pequeña descripción general del requisito.
- **Prioridad:** hace referencia al orden temporal en que debe aplicarse el requisito. “Alta” indica lo antes posible y “baja” indica que no es muy urgente.
- **Necesidad:** importancia o necesidad del requisito en el sistema.
- **Estabilidad:** indica si el requisito se mantendrá igual a lo largo del proyecto, ó si por el contrario, podrá ser modificado.
- **Verificable:** indica si es verificable o no que dicho requisito se ha cumplido.
- **Descripción:** especificación del requisito mediante texto explicativo.
- **Fuente:** indica el origen del requisito. Puede ser un documento, otro requisito, una persona.

En las páginas siguientes se van a detallar los requisitos software del sistema organizado en las categorías anteriormente descritas:

### 3.5.1. Requisitos de Software Funcional

RSF-01			
<b>Título</b>	Iniciar sesión		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-03
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Un usuario podrá iniciar sesión en el sistema proporcionando sus datos de acceso.		

Tabla 73: RSF-01

RSF-02			
<b>Título</b>	Añadir un nuevo usuario al sistema		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-02, RUC-08
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Se permite al administrador dar de alta a un usuario a partir de los datos personales que este ofrezca.		

Tabla 74: RSF-02

RSF-03			
<b>Título</b>	Modificar los datos personales de un usuario		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-02, RUC-14, RUC-15
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	A partir del identificador de usuario se ofrece la posibilidad al administrador de modificar los datos personales de un usuario de forma que se le muestre los datos que se le permite modificar.		

Tabla 75: RSF-03

RSF-04			
<b>Título</b>	Borrar los datos personales de un usuario		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-02, RUC-10, RUC-11
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	A partir del identificador de usuario se ofrece la posibilidad al administrador de eliminar a un usuario de la aplicación		

Tabla 76: RSF-04

RSF-05			
<b>Título</b>	Consultar los datos personales de un usuario		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-02, RUC-05, RUC-07
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	A partir del identificador de usuario se ofrece la posibilidad al administrador de obtener los datos personales de un usuario de forma que este pueda acceder a la información del usuario.		

Tabla 77: RSF-05

RSF-06			
<b>Título</b>	Añadir un nuevo libro al sistema		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-01, RUC-09
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Se permite al administrador dar de alta a un libro a partir de los datos que se obtengan de este.		

Tabla 78: RSF-06



RSF-07			
<b>Título</b>	Modificar los datos de un libro		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-03, RUC-16, RUC-17
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	A partir del identificador de un libro se ofrece la posibilidad al administrador de modificar los datos de este de forma que se le muestre los datos que se le permite modificar.		

Tabla 79: RSF-07

RSF-08			
<b>Título</b>	Borrar los datos de un libro		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-03, RUC-12, RUC-13
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	A partir del identificador de libro se ofrece la posibilidad al administrador de eliminarlo de la aplicación		

Tabla 80: RSF-08

RSF-09			
<b>Título</b>	Consultar los datos de un libro		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-03, RUC-04, RUC-06
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	A partir del identificador de un libro, se ofrece la posibilidad a cualquier tipo de usuario de obtener los datos de este, de forma que este pueda acceder a la información del libro en cuestión.		

Tabla 81: RSF-09

RSF-10			
<b>Título</b>	Préstamo de un libro		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-01, RUC-04, RUC-18, RUC-19
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	A partir del identificador de un libro, un usuario podrá obtener el préstamo de un libro. Se le devolverá un identificador de préstamo con el cuál podrá terminar de realizar el préstamo.		

Tabla 82: RSF-10

RSF-11			
<b>Título</b>	Devolución de un libro		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-01, RUC-04, RUC-20, RUC-21
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	A partir del identificador de un libro, y del identificador de préstamo, se podrá devolver un libro.		

Tabla 83: RSF-11

RSF-12			
<b>Título</b>	Posibilidad de cambio de contraseña		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-22
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	A partir de la contraseña actual del usuario, es posible cambiar de contraseña. Para ello será necesario introducir a parte de la contraseña actual, la contraseña nueva y volver a repetir la contraseña nueva por motivos de seguridad.		

Tabla 84: RSF-12

RSF-13			
<b>Título</b>	Posibilidad de visualizar el historial de préstamos		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-23
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Un usuario que haya accedido al sistema será capaz de visualizar los 5 últimos préstamos que haya efectuado.		

Tabla 85: RSF-13

RSF-14			
<b>Título</b>	Posibilidad de visualizar préstamos actuales		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-24
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Un usuario que haya accedido al sistema será capaz de visualizar los préstamos que tenga en la actualidad.		

Tabla 86: RSF-14

RSF-15			
<b>Título</b>	Posibilidad de visualizar la información personal		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-25
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Un usuario que haya accedido al sistema será capaz de visualizar su propia información, mostrando a parte de los datos, si tiene préstamos en la actualidad así como si está penalizado.		

Tabla 87: RSF-15

RSF-16			
<b>Título</b>	Aviso de préstamos máximos		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-26
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	La aplicación avisará al usuario de que no es posible realizar el préstamo del libro mediante un mensaje en el dispositivo móvil.		

Tabla 88: RSF-16

RSF-17			
<b>Título</b>	Aviso de penalización		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-27
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	La aplicación avisará al usuario de que está penalizado mostrando un mensaje en el dispositivo móvil. En el mensaje le avisará del tiempo que tiene de penalización		

Tabla 89: RSF-17

RSF-18			
<b>Título</b>	Avisos mediante correo electrónico		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-28, RUC-29
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	La aplicación enviará al usuario mensajes mediante correos electrónicos, avisando de los préstamos, penalizaciones y vencimiento de préstamo		

Tabla 90: RSF-18

RSF-19			
<b>Título</b>	Dos aplicaciones distintas		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-30
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	<p>En la aplicación habrá dos tipos de aplicaciones para usuarios con diferentes roles:</p> <ul style="list-style-type: none"> <li>• <b>Aplicación cliente administrador:</b> Será para personas con roles de administrador, esta aplicación permite realizar préstamos y devoluciones.</li> <li>• <b>Aplicación cliente general:</b> Esta orientada a personas que no tienen el rol de administrador. Es una versión en la cual no se pueden realizar préstamos y devoluciones.</li> </ul>		

Tabla 91: RSF-19

RSF-20			
<b>Título</b>	Varios tipos de usuarios en la aplicación		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-31, RUC-32, RUC-33
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	<p>En la aplicación habrá distintos tipos de usuarios que tendrán distintos tipos de préstamos dado su rol.</p> <ul style="list-style-type: none"> <li>• <b>Alumno:</b> Tendrá un máximo de dos libros que se le prestará durante una semana cada uno. En caso de penalización será de un día por ejemplar no entregado.</li> <li>• <b>Profesor:</b> Tendrá un máximo de cuatro libros que se le prestará durante dos semanas cada uno. En caso de penalización será de un día por ejemplar no entregado.</li> </ul>		

Tabla 92: RSF-20

RSF-21			
<b>Título</b>	Posibilidad de renovar un préstamo		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-34
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Un usuario que haya accedido al sistema será capaz de renovar un préstamo que tenga en su poder, siempre que esté a tres días de su finalización.		

Tabla 93: RSF-21

RSF-22			
<b>Título</b>	Posibilidad de reservar un libro		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUC-35
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Un usuario que haya accedido al sistema será reservar un libro siempre que esté disponible guardándose la reserva durante tres días.		

Tabla 94: RSF-22

### 3.5.2. Requisitos no funcionales

RSNF-01			
<b>Título</b>	Inicio de sesión diferente dado el tipo de usuario		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-01
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Dependiendo del tipo de usuario, se podrá acceder a la aplicación del bibliotecario en el caso del administrador, o la del cliente para el resto de usuario, no pudiendo acceder cada uno a otra aplicación.		

Tabla 95: RSNF-01

RSNF-02			
<b>Título</b>	Integridad de la base de datos		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-02
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	El sistema tendrá mecanismos de protección de la integridad de los datos personales de cada usuario registrado en el sistema, de forma que no será posible acceder a los datos personales de cada usuario si no se tienen los permisos suficientes.		

Tabla 96: RSNF-02

RSNF-03			
<b>Título</b>	Idioma de la aplicación		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-03
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	El idioma que por el momento se tendrá en cuenta en la aplicación será el Castellano, pudiéndose extender fácilmente a otros idiomas en el futuro si la aplicación lo requiriese.		

Tabla 97: RSNF-03

RSNF-04			
<b>Título</b>	Tipo de dispositivos para la aplicación		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-05
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Únicamente la aplicación funcionará con dispositivos Android 2.1 o posterior.		

Tabla 98: RSNF-04

RSNF-05			
<b>Título</b>	Tiempo de espera mínimo		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-04
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	El sistema debe responder en menos de 10 segundos a cada petición realizada desde el dispositivo móvil.		

Tabla 99: RSNF-05

RSNF-06			
<b>Título</b>	Conexión a Internet a través de redes Wifi		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-06
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Para poderse conectar con el servicio web, el dispositivo tendrá que poder conectarse a internet a través de redes Wifi.		

Tabla 100: RSNF-06

RSNF-07			
<b>Título</b>	Conexión a Internet a través de redes 3G.		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-06
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Para poderse conectar con el servicio web, el dispositivo tendrá que poder conectarse a internet a través de redes 3G.		

Tabla 101: RSNF-07

RSNF-08			
<b>Título</b>	Disponibilidad de Cámara de Fotos.		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-12
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Para poderse usar el escanedor de códigos inteligentes, el dispositivo tendrá que disponer de una cámara de fotos para poder capturar los códigos inteligentes.		

Tabla 102: RSNF-08

RSNF-09			
<b>Título</b>	Registro a través del administrador		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-07
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Como medida de seguridad, únicamente el administrador podrá registrar nuevos usuarios al sistema, impidiendo el registro automático		

Tabla 103: RSNF-09

RSNF-10			
<b>Título</b>	Almacenar datos de sesión en el dispositivo móvil		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-08
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	La aplicación almacenará los datos de inicio de sesión con el fin de evitar al usuario introducirlos cada vez que acceda a las distintas funcionalidades.		

Tabla 104: RSNF-10



RSNF-11			
<b>Título</b>	Mensajes en el dispositivo móvil		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-09, RUR-10
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	Con fin de informar al usuario, la aplicación enviará mensajes emergentes con el fin de avisar en todo momento de lo que ocurre. Dichos mensajes serán de información sobre el número máximo de préstamos y penalización por entrega tarde de libros.		

Tabla 105: RSNF-11

RSNF-12			
<b>Título</b>	Tipo de etiquetas inteligente a escanear		
<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Fuente</b>	RUR-11
<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional		
<b>Estabilidad</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No	<b>Verificable</b>	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
<b>Descripción</b>	El sistema únicamente podrá escanear etiquetas inteligentes del tipo: <ul style="list-style-type: none"><li>• Code 2D</li><li>• Code 39</li></ul>		

Tabla 106: RSNF-12





## 4. Diseño



A continuación, en el siguiente capítulo se va a mostrar el diseño arquitectónico en el que se aprecia el diseño detallado del sistema. Además se ofrece una explicación de cada una de las decisiones tomadas a la hora de realizar el diseño.

#### **4.1. Arquitectura del sistema**

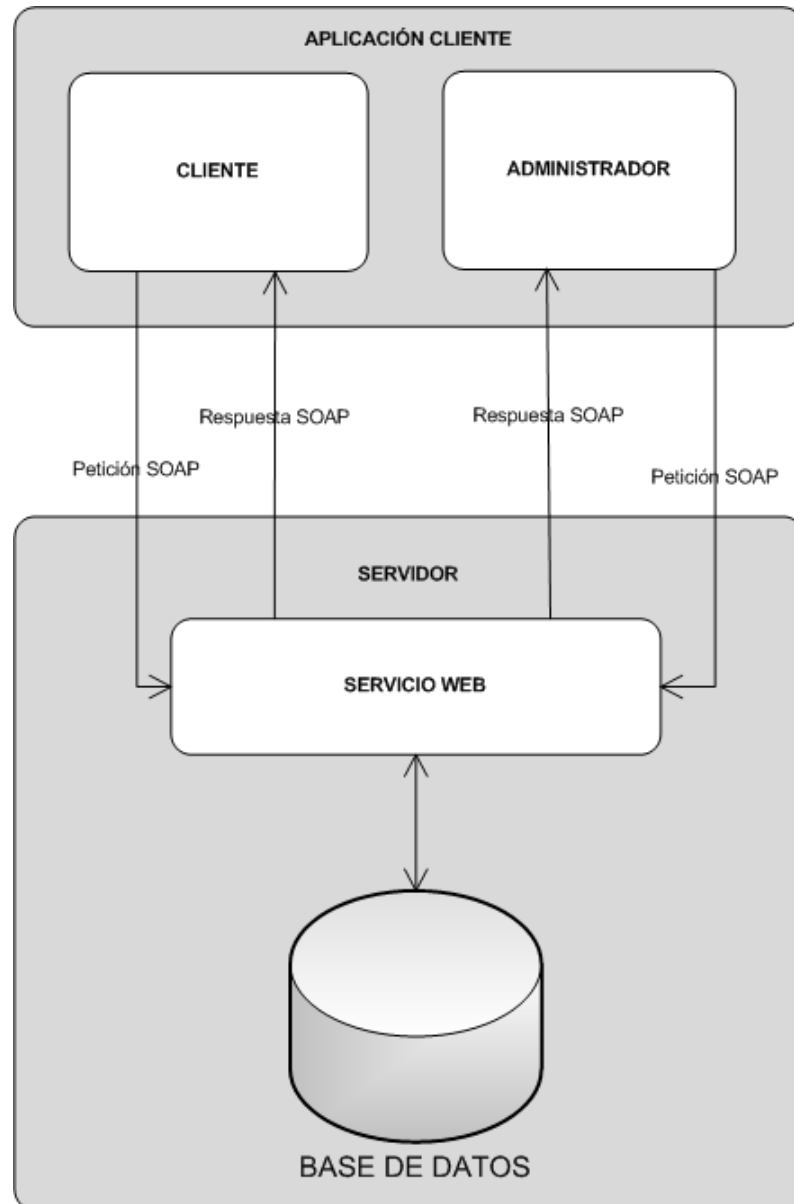
En el siguiente apartado se va a mostrar el diseño de la arquitectura del sistema, en el cuál se puede apreciar que se ha desarrollado un sistema separado en servidor y cliente. La parte del servidor será implementada mediante la tecnología .NET, mientras que la parte del cliente será a través de Android.

Como se puede apreciar en la figura siguiente, en el sistema creado se puede apreciar varios componentes. En un primer lugar, se observa una arquitectura Cliente-Servidor. La parte del cliente será toda aquella que será usada por cualquier tipo de usuario final, que en este caso serán las distintas aplicaciones Android. En el caso contrario, el servidor será el resto de componentes, es decir, serán los módulos correspondientes al servicio web y la base de datos en la cuál se almacena toda la información concerniente a los usuarios y la información almacenada de la biblioteca.

En cuanto a la parte del Cliente, tenemos que apreciar que esta diferenciada en dos subcomponentes distintos, que se corresponden con cada una de las aplicaciones distintas que pueden manejar los usuarios.

El apartado concerniente al Servidor se refiere a otros dos subcomponentes tales como el servicio web que se usará como puente entre las aplicaciones Android y la base de datos, a través de mecanismos como SOAP.

Más adelante se explican con más detalle cada uno de los módulos que compone esta arquitectura, así como la relación existente entre ellos. A continuación se aprecia un diagrama que muestra cada uno de los componentes que compone la arquitectura del sistema.



**Ilustración 11: Arquitectura del sistema**

El sistema creado, a parte de usar una arquitectura Cliente-Servidor se basará a su vez en el patrón arquitectónico Modelo Vista Controlado (MVC), debido a que como se ha visto anteriormente, el sistema tiene dos partes diferenciadas que son el modelo Cliente-Servidor, en la cuál se puede integrar internamente el patrón MVC.

Para aclarar dudas que puedan surgir sobre cada uno de los dos modelos arquitectónicos, a continuación se procede a explicar cada uno de ellos, con el objetivo de aclarar dudas sobre el funcionamiento del sistema.

#### 4.1.1. Modelo Vista Controlador (MVC)

El *Modelo Vista Controlador* es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario (GUI), y la lógica de negocio en tres componentes distintos. El patrón MVC, se aprecia frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.

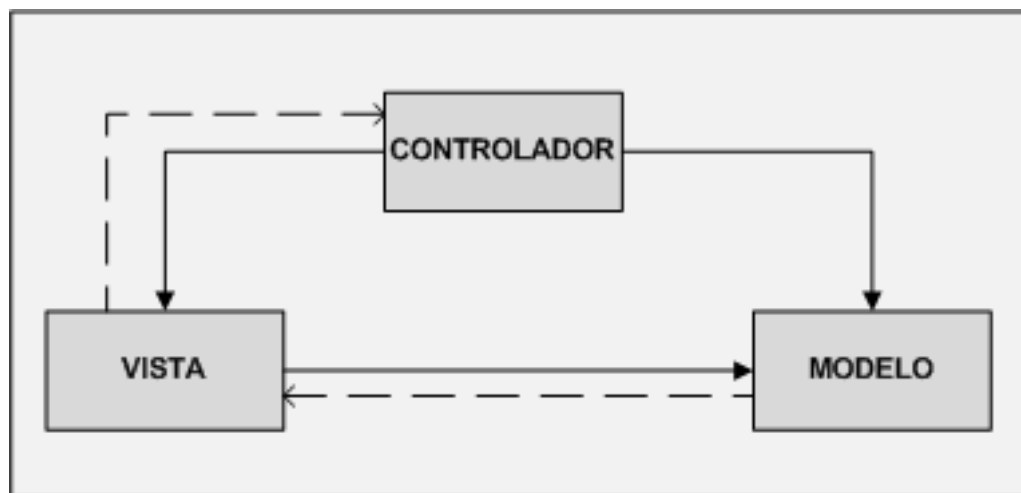


Ilustración 12: Arquitectura MVC

De una forma más simple, se puede el patrón MVC como un patrón arquitectural que describe una forma de desarrollar aplicaciones software separando los componentes en tres grupos, los cuales explicamos a continuación:

- **Modelo:** es la representación específica de la información con la cual el sistema opera. En resumen, el modelo se limita a lo relativo de la vista y su controlador facilitando las presentaciones visuales complejas. El sistema también puede operar con más datos no relativos a la presentación, haciendo uso integrado de otras lógicas de negocio y de datos afines con el sistema modelado.
- **Vista:** presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario (GUI).
- **Controlador:** responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente a la vista.

#### 4.1.2. Modelo Cliente-Servidor

Es un modelo en el cual las tareas se reparten entre los servidores y los llamados clientes. Un cliente realiza peticiones a otro programa y el servidor es el que se encarga de enviarle la respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre un solo ordenador, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

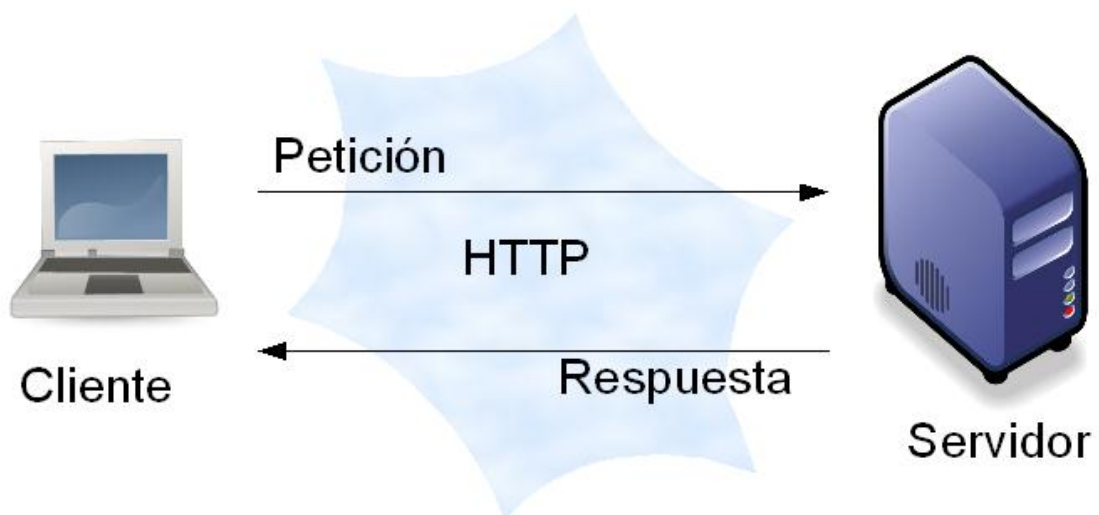


Ilustración 13: Arquitectura Cliente-Servidor

- **Cliente:** maneja todas las funciones relacionadas con la manipulación y despliegue de datos, por lo que están desarrollados sobre plataformas que permiten construir interfaces gráficas de usuario (GUI), además de acceder a los servicios distribuidos en cualquier parte de una red.
- **Servidor:** es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Al proceso servidor se le conoce con el término back-end. Normalmente el servidor maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos.



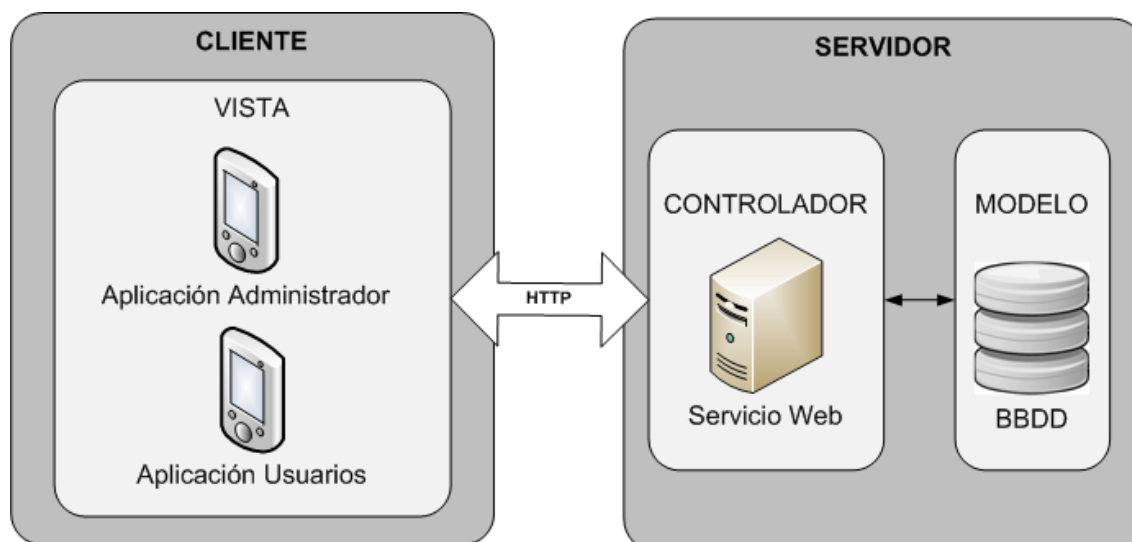
El sistema se puede apreciar, que se comporta como un Cliente-Servidor debido a que es un sistema distribuido en cual los clientes realizan peticiones mediante SOAP en los dispositivos móviles Android, mientras que la parte del servidor es el servicio web junto con la base de datos que reciben peticiones a través de los dispositivos móviles y les devuelven respuestas con la información solicitada por los dispositivos.

El sistema final se compone de un servidor que almacena los datos y gestiona las peticiones y de dos tipos de aplicaciones cliente, una para el administrador de la biblioteca y otra para los múltiples usuarios finales, que son los que necesitarán realizar peticiones al servidor con el fin de obtener unos datos.

En cuanto a la arquitectura MVC el sistema nos indica que la parte de la Vista serán los dispositivos móviles, que en este caso son los múltiples clientes Android. Por el contrario, la parte del Servidor contiene al Controlador que será el Servicio Web que se encarga de realizar todas las peticiones y respuestas entre la base de datos y los clientes, y al Modelo que va a ser la base de datos que almacena la información privada de los usuarios, así como la de la biblioteca.

Por lo tanto podemos apreciar que la arquitectura del sistema es un Modelo Vista Controlador modificado, debido a que el cliente no tiene permitido acceder al Modelo directamente por cuestiones de seguridad, por lo que obligatoriamente tendrá que acceder al Modelo, a través del Controlador. La decisión de usar un Modelo Vista Controlador modificado, en lugar del tradicional es como se ha visto por cuestiones seguridad principalmente.

Debido a esta decisión, se han creado un conjunto de funciones en el Controlador capaces de gestionar las peticiones que la Vista le realiza para obtener datos del Modelo.



**Ilustración 14: Arquitectura del sistema completa**

Según el diagrama mostrado anteriormente, los pasos a seguir para poder enviar peticiones al servidor y obtener así respuestas son los que se muestran a continuación:

- El usuario desde una de las aplicaciones móviles (Vista) envía una petición al servicio Web.
- La aplicación crea una solicitud HTTP y la envía a la URL donde está alojado el servidor, encapsulada en una petición SOAP.
- La petición SOAP llega al Servicio Web (Controlador) que la gestiona y dependiendo del tipo de petición, hace una consulta a la base de datos (Modelo).
- El servidor crea una respuesta HTTP que incluye la respuesta SOAP del Servicio Web, enviándolo a la dirección desde la que recibió la petición, que en este caso es el dispositivo móvil que envió la petición SOAP.
- En el dispositivo móvil se recibe la respuesta del servidor, decodificándola y procesando los datos obtenidos.
- El dispositivo muestra los datos recibidos.

## **4.2. Diseño detallado**

El diseño detallado muestra el diseño de cada interfaz de la aplicación de usuario (GUI), así como del diagrama de clases Android, y del diseño de la base de datos. El diseño del sistema se corresponde como se ha visto anteriormente a la arquitectura Cliente-Servidor en la que está compuesto por un MVC modificado tal y como se explicó en el apartado anterior.

### **4.2.1. Diseño de interfaces de usuario**

En este apartado se va a explicar de qué forma se ha llevado a cabo el diseño de las interfaces de usuario, ya que como la aplicación principal es la que se aloja dentro del dispositivo Android y será usada por los distintos usuarios finales, pues se ha decidido darle algo de importancia. Por este motivo se considera uno de los puntos más relevantes en la implementación de la aplicación, especialmente si se trata de una aplicación móvil como es este el caso.

Es necesario que una aplicación como la que se ha presentado a lo largo de este proyecto, tenga que ser buena gráficamente. El motivo es simple, y es que sin un buen diseño que sea capaz de atraer al público objetivo, o que sea complicada de usar, terminará por no usarse y caerá en el olvido.

A la hora de diseñar las distintas pantallas de la aplicación, se ha de tener en cuenta, una profunda comprensión de cómo el usuario va a usar la aplicación. Por este motivo se ha realizado un análisis de requisitos previo en capítulos anteriores, en el cual se ha tenido en cuenta la interfaz del usuario desde el punto que contenga toda la funcionalidad necesaria para el usuario final.

Por estos motivos, el objetivo de este apartado es aclarar que se ha realizado una interfaz sencilla y cómoda para los usuarios, debido a que si el usuario no termina de entender la interfaz se pierde mucho tiempo en realizar cualquier tarea perdiendo eficiencia, por lo que es necesario realizar una interfaz cómoda y sencilla aplicando criterios y técnicas de usabilidad, no sólo porque al usuario le sea agradable, sino porque a la larga se terminará ahorrando tiempo. Debido a todo lo comentado anteriormente se ha realizado una interfaz de usuario basada en la interfaz normal de un dispositivo Android de que es sencilla y eficaz.

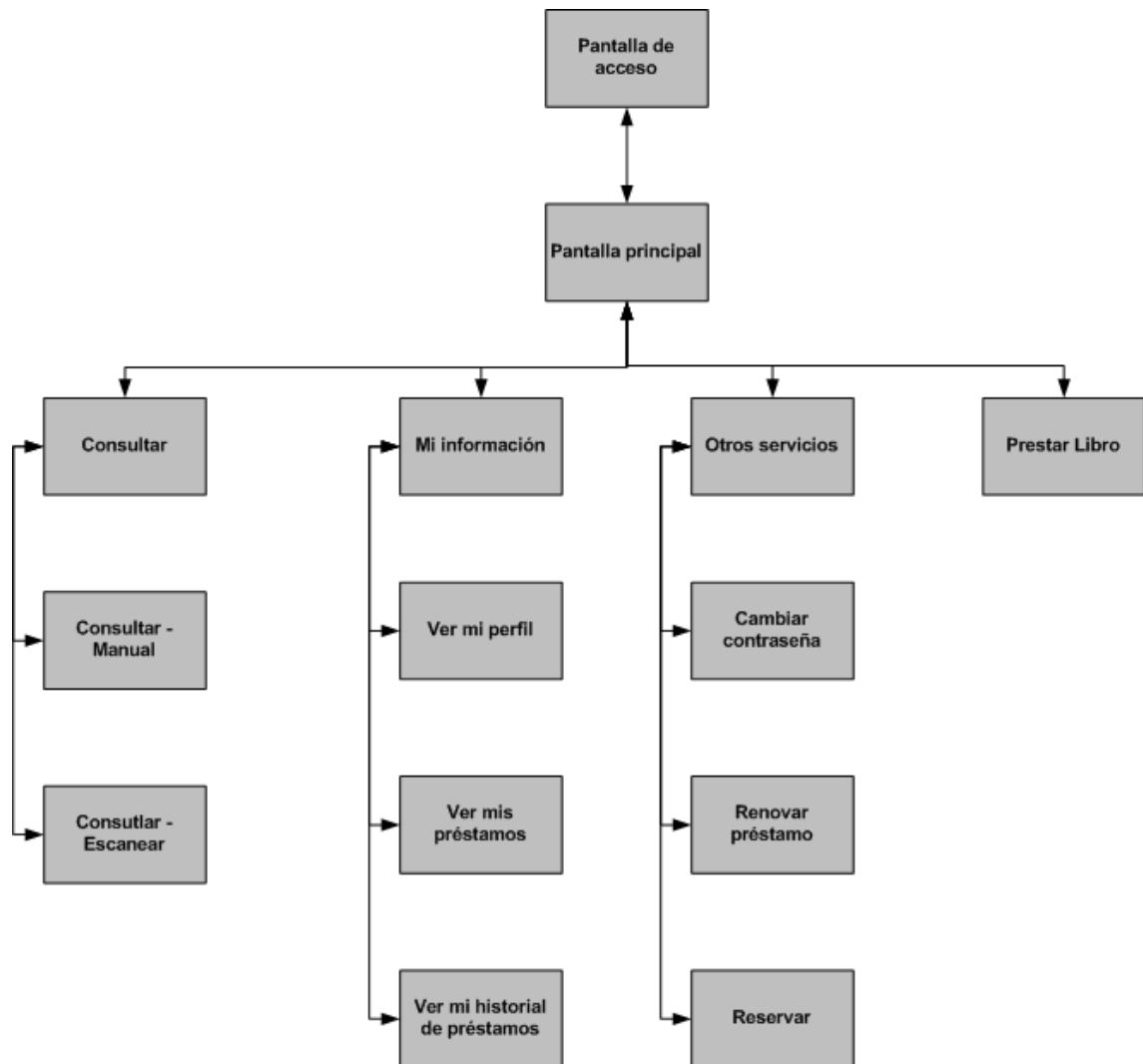
En el *Anexo B* se van a mostrar las distintas interfaces utilizadas, pero antes de llegar a dicho apartado se describen varios puntos en los que se ha basado el diseño de las interfaces de usuario:

- Las distintas pantallas de cada aplicación seguirán un aspecto homogéneo de forma que se mantenga un conjunto coherente de colores, fuentes y estilos para la aplicación.
- Cada aplicación seguirá su propio patrón de colores, fuentes y estilos de forma que se identifique a golpe de vista que aplicación se está ejecutando basándose en los estilos de cada aplicación.
- Acceso rápido a los distintos contenidos de cada aplicación.
- Diseñar la aplicación del administrador mediante pestañas de forma que no se sobrecargue el contenido por el exceso de botones que contiene.
- Seguir las recomendaciones dadas por Google que se presentan en la página dedicada a los desarrolladores de Android.
- Se presupone que los usuarios no son grandes expertos en el manejo de aplicaciones por lo que se ha intentado reflejar en la medida de lo posible una navegación de pantallas sencilla y clara. Para ello se han usado botones grandes y visibles de colores muy llamativos con son el naranja y el azul.
- Se van a utilizar textos aclaratorios en cada una de las pantallas, con el objetivo de saber en que parte de la aplicación nos encontramos.

A continuación se va a mostrar un diagrama de navegación para cada una de las dos aplicaciones Android.

### ***Diagrama de navegación - Aplicación Cliente***

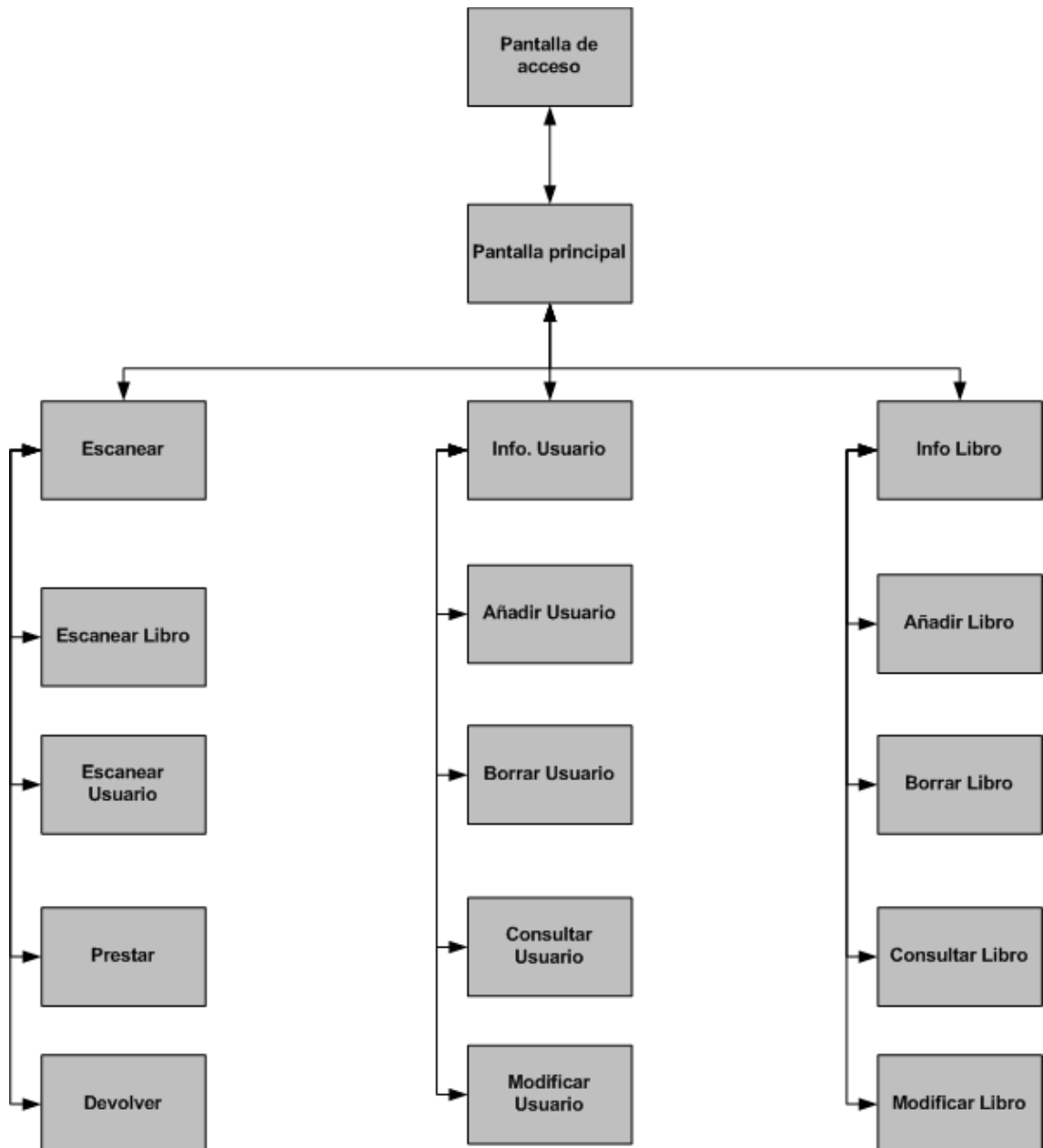
En este apartado se va a mostrar un diagrama de navegación de la aplicación cliente, con el objetivo de mostrar la navegabilidad entre las distintas pantallas que contiene esta aplicación, para este tipo de usuarios.



**Ilustración 15: Diagrama de navegación – aplicación cliente**

### ***Diagrama de navegación - Aplicación Administrador***

En este apartado se va a mostrar un diagrama de navegación de la aplicación administrador, con el objetivo de mostrar la navegabilidad entre las distintas pantallas que contiene esta aplicación, para que el administrador pueda realizar sus tareas de control y gestión del sistema.



**Ilustración 16: Diagrama de navegación – aplicación administrador**

### 4.2.2. Aplicaciones Android

En esta sección de la memoria se entrará en los detalles arquitectónicos de la aplicación desarrollada. Por este motivo se incluye un diagrama de clases de cada aplicación Android obteniendo una visión global de la aplicación.

En los diagramas de clases que se muestran a continuación hay que tener en cuenta que son muy similares entre sí ya que ciertas funcionalidades básicas son comunes a ambas aplicaciones, pero hay que diferenciar varios aspectos que las diferencian entre sí, los cuales se pueden ver a continuación.

Como se aprecia en los diagramas siguientes, cada clase está representada por una vista. Esto es así ya que al diseñar una aplicación Android es necesario que cada pantalla vaya asociada a una vista distinta. Es por este motivo, por lo que aparecen tantas clases, pero como no se va a realizar ningún cálculo, ni se va a almacenar ningún dato en los dispositivos, pues simplemente las clases muestran las distintas vistas de la aplicación. Se pueden apreciar distintas pantallas que se reutilizan, que coinciden que son las que muestran algún tipo de información al usuario en el dispositivo.

Las clases contienen atributos y métodos que permiten conectarse al servicio web mediante el protocolo SOAP, a través de peticiones y respuestas HTTP, así como de otras funciones que requieren de comunicaciones de red, ya que es necesario debido a que cada aplicación se mantienen conectadas a un servidor, teniendo peticiones y respuestas constantes con el servidor.

A continuación se procede a explicar cada una de las clases de forma aclaratoria de los diagramas:

#### *Clases comunes*

- **ControlMenu:** esta clase es la principal de cada aplicación, desde la cual se gestionan casi todos los elementos de cada aplicación. Muestra un menú que contiene distintos botones que permiten acceder a las distintas funcionalidades. En la aplicación administrador, además contiene tres pestañas desde la que se puede acceder a cada botón.
- **ControlLogin:** esta clase permite acceder a las distintas aplicaciones permitiendo al usuario, poder introducir su nombre usuario y contraseña a través del servicio web.
- **Libro:** Esta clase permite gestionar los distintos libros que se recuperan de la base de datos organizando toda la información que provenga de las respuestas SOAP, así como poder calcular el ISBN de un libro a partir de su código de barras.
- **UsuarioCliente:** esta clase permite recuperar los datos del usuario del servicio web y mostrarlos de forma coherente en la aplicación.

- **ChangePassword:** esta clase es la encargada de poder cambiar una contraseña a un usuario específico. Para ello es necesario que el usuario introduzca su contraseña antigua, así como la nueva y por motivos de seguridad, volver a repetir la misma contraseña que usará en un futuro.
- **UserConsulted:** permite consultar a través del servicio web, los datos de un usuario específico. Es necesario introducir el identificador de usuario para poder obtener sus datos.
- **BookConsulted:** permite consultar a través del servicio web, los datos de un libro específico. Es necesario introducir el identificador del libro para poder obtener sus datos.
- **UserBorrow:** establece a un usuario que permite autoprestarse un libro en el caso de la aplicación del cliente, y de realizar un préstamo a un cliente en el caso del administrador.
- **BookBorrowManual:** permite a un usuario realizar un préstamo a través del servicio web usando como identificador del libro a prestar el ISBN que será introducido por el mismo usuario a mano.
- **BookBorrowScan:** permite a un usuario realizar un préstamo a través del servicio web usando como identificador del libro a prestar el ISBN que será leído a través del ScanCode que contiene la aplicación leyendo su código de barras.
- **ModifyUserDataSend:** permite modificar los datos de un usuario a la hora de realizar una devolución a través del servicio web.
- **ModifyBookDataSend:** permite modificar los datos de un libro a la hora de realizar una devolución a través del servicio web.

### ***Clases propias de la aplicación Administrador***

- **AddUser:** permite añadir un usuario al administrador. Para ello es necesario introducir todos los datos que se requieren para que se añada un usuario.
- **DeleteUser:** permite borrar un usuario al administrador. Para ello es necesario introducir el identificador de usuario para que el servicio web lo busque en la base de datos.
- **ModifyUser:** permite modificar un usuario al administrador. Para ello es necesario introducir el identificador de usuario para que el servicio web lo busque en la base de datos, y devuelva todos los datos del usuario que el

administrador modificará. Una vez realizados los cambios, se guardan de nuevo en la base de datos.

- **AddBook:** permite añadir un libro al administrador. Para ello es necesario introducir todos los datos que se requieren para que se añada un libro.
- **DeleteBook:** permite borrar un libro al administrador. Para ello es necesario introducir el identificador del libro para que el servicio web lo busque en la base de datos.
- **ModifyBook:** permite modificar un libro al administrador. Para ello es necesario introducir el identificador de libro para que el servicio web lo busque en la base de datos, y devuelva todos los datos del libro que el administrador modificará. Una vez realizados los cambios, se guardan de nuevo en la base de datos.
- **ModifyUserDataReturn:** permite modificar los datos de un usuario a la hora de realizar un préstamo a través del servicio web.
- **ModifyBookDataReturn:** permite modificar los datos de un libro a la hora de realizar un préstamo a través del servicio web.

### *Clases propias de la aplicación Cliente*

- **DeleteBook:** permite borrar un libro al administrador. Para ello es necesario introducir el identificador del libro para que el servicio web lo busque en la base de datos.
- **ModifyBook:** permite modificar un libro al administrador. Para ello es necesario introducir el identificador de libro para que el servicio web lo busque en la base de datos, y devuelva todos los datos del libro que el administrador modificará. Una vez realizados los cambios, se guardan de nuevo en la base de datos.
- **Renovar:** permite gestionar una renovación por parte de un cliente. Para ello es necesario introducir el identificador del libro que actualmente está prestado, introduciéndolo a falta de tres días para su caducidad.
- **Reservar:** permite realizar una reserva de un libro que actualmente ningún usuario tiene prestado. Si el libro no se ha recogido en tres días, vuelve a estar libre.

A continuación se muestra el diagrama de clases:



## Aplicación Administrador

El diagrama de clases que se aprecia en este apartado es el concerniente a la aplicación del administrador del sistema.

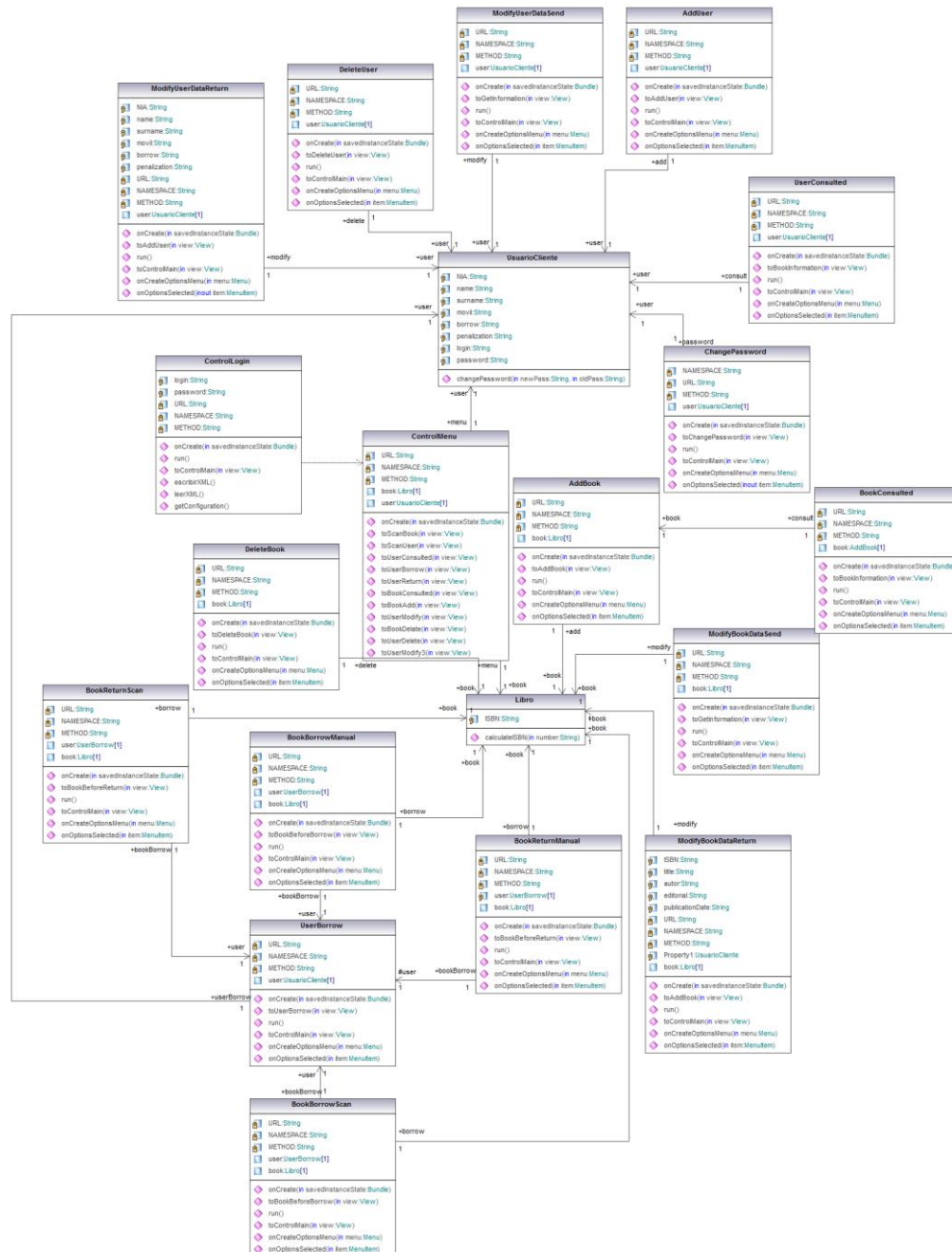


Ilustración 17: Diagrama de clases de la aplicación del administrador

## Aplicación Cliente

El diagrama de clases que se aprecia en este apartado es el concerniente a la aplicación del usuario final del sistema.

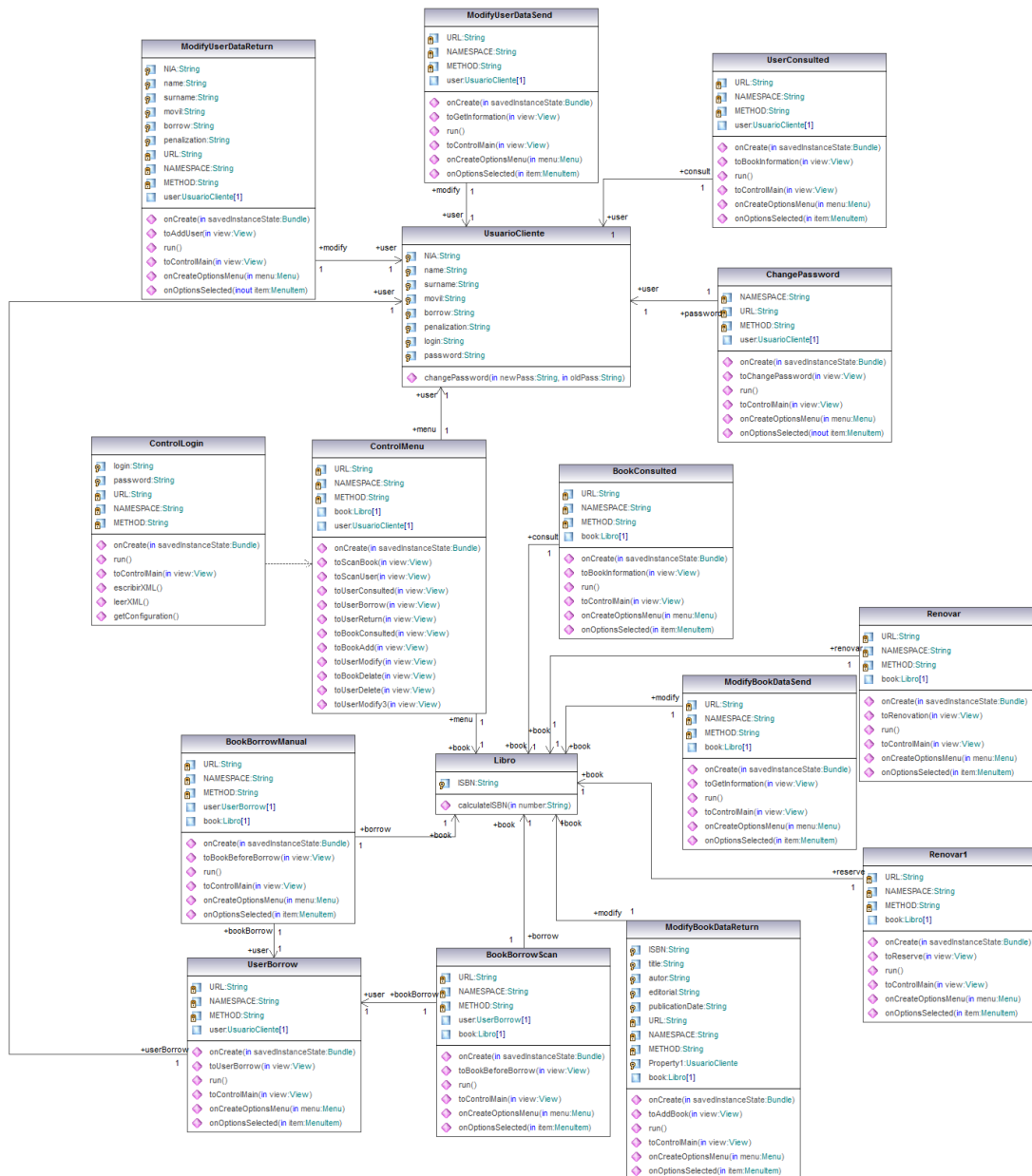


Ilustración 18: Diagrama de clases de la aplicación del usuario

### 4.2.3. Servicio Web

A continuación se entrará en los detalles arquitectónicos del servicio web de forma que se explique como se ha llevado a cabo su desarrollo. Por este motivo se incluye un diagrama de clases del servicio web desarrollado en .NET obteniendo una visión global de la aplicación.

En el diagrama de clases que se muestran a continuación hay que tener en cuenta que la clase Servicio Web es la principal de la aplicación y es la que se conectará a la base de datos y a las distintas aplicaciones cliente, debido a que esta es la clase que contiene las cabeceras de los métodos que permiten establecer la comunicación propios del servicio web. Esta clase contiene las cabeceras de todos los métodos que intervienen en el servicio web de forma que cuándo se llama a uno de estos métodos por parte de una aplicación cliente, pasa por esta clase, yendo directamente a la clase en la cuál se implementa por completo el método.

El resto de clases las cuales se pueden apreciar en el diagrama que se muestra a continuación contienen los métodos desarrollados ya que como se comentaba anteriormente la clase Servicio Web únicamente contiene las cabeceras de los métodos, junto su respectiva llamada a la clase que contiene el desarrollo del método en cuestión.

Tal y como se puede apreciar a continuación, el servicio web es el controlador dentro del patrón MVC ya explicado. Esto es debido a que es quien interviene entre las aplicaciones cliente que solicitan información, y la base de datos que la contiene y la cuál es proporcionada por el servicio web.

Los métodos de la clase *ServicioWeb* contienen una cabecera que permiten conectarse a la base de datos, así como a las aplicaciones cliente mediante el protocolo SOAP, a través de peticiones y respuestas HTTP.

A continuación se muestra el diagrama de clases propio del servicio web:



Ilustración 19: Diagrama de clases del servicio web

#### 4.2.4. Base de datos

La base de datos de la aplicación compone el almacén de datos de todo el sistema que almacena en su interior tanto a los usuarios como a la información concerniente a los libros. La información que se gestiona en su interior es tramitada por el servicio web. Dicha base de datos contiene las siguientes tablas:

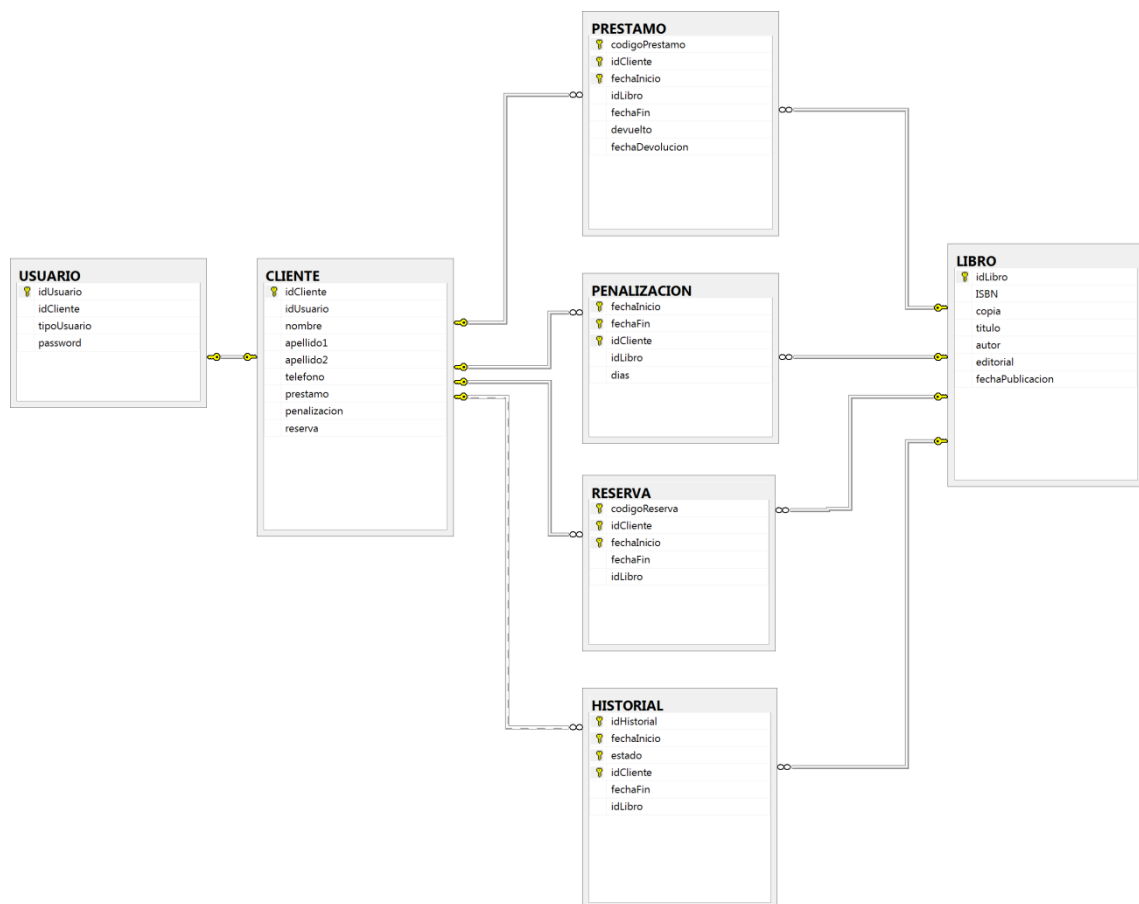


Ilustración 20: Base de datos

En la base de datos de la figura anterior, hay que tener en cuenta que cada cambio que se realiza es de forma automatizada, debido a que es el servicio web el que la gestiona, es decir, no hay ningún tipo de interacción humana con la base de datos sino, que mediante procedimientos del servicio web, se pueden desde añadir información a las distintas tablas hasta la actualización de dicha información, así como el borrado de alguna inserción.



# 5. Implementación y Pruebas





En el siguiente apartado se explica el procedimiento seguido y los detalles de la implementación. Además se incluye un conjunto de pruebas realizadas para la verificación del correcto funcionamiento del sistema.

## 5.1. Implementación

A continuación se muestran los diferentes pasos realizados a la hora de realizar la implementación del sistema, así como los diferentes problemas que han surgido y como se han solucionado.

Para poder implementar el sistema completo, fue necesario inicialmente tener claro que se iba a realizar. La siguiente lista muestra en que orden se han implementado cada uno de los elementos del sistema.

- **Creación del servidor:** se crea un servidor IIS desde el que se puede crear posteriormente el servicio web y la base de datos que alojará, con el fin, de que se puedan obtener datos a partir de la aplicación cliente.
- **Implementación del servicio web:** se crea un Servicio Web bajo la tecnología .Net. Este servicio web está escrito bajo el lenguaje de programación C#. Se incluyen todas las funciones necesarias para comunicarse correctamente a través de peticiones realizadas.
- **Creación de la base de datos:** se crea una base de datos alojada en el servidor. Esta base de datos únicamente podrá ser accedida por el servicio web, con el objetivo de mantener la integridad de los datos y no puedan ser accedidos si no se tienen los suficientes permisos.
- **Implementación de las pantallas Android:** este paso trata de implementar la vista del usuario de las distintas pantallas de cada aplicación que se va a crear.
- **Gestión de las comunicaciones:** una vez creado el servidor y el servicio web, es necesario establecer mediante SOAP la comunicación, creando un conjunto de funciones que permitan tanto realizar solicitudes a partir de una URL como gestionar el resultado de las mismas. Por este motivo se implementan un conjunto de controladores que aseguren la correcta comunicación.
- **Servicio de correo:** este paso trata de implementar un servicio de correo con el fin de enviar un correo electrónico a aquellos usuarios que realicen un evento tal como el préstamo y devolución de un libro entre otros.
- **Servicio de pruebas unitarias:** una vez que se han creado todos los módulos de la parte de implementación, se crean pruebas unitarias para probar todo lo relativo a la base de datos y de cómo interacciona con el servicio web

A continuación se detallan con una mayor profundidad, cada uno de los pasos vistos anteriormente:

### 5.1.1. Creación del servidor

Inicialmente se procede a crear el servidor IIS del sistema, desde el cual se almacenará el servicio web que se va a crear a continuación y su base de datos desde la que se almacenarán todos los datos necesarios del sistema. Este servidor se ha creado, tal y como se comentó anteriormente debido a su simplicidad y su eficacia, permitiendo usar herramientas potentes tales como el Framework de Microsoft .NET.

Para instalar el servidor IIS, es necesario disponer de un sistema operativo Windows 7 y activarlos desde una de sus opciones. Este sistema operativo permite de una forma fácil, que el usuario disponga de un servidor completo para poder almacenar y gestionar datos. A continuación se procede a configurarlo para poder obtener la funcionalidad requerida para implementar el sistema.

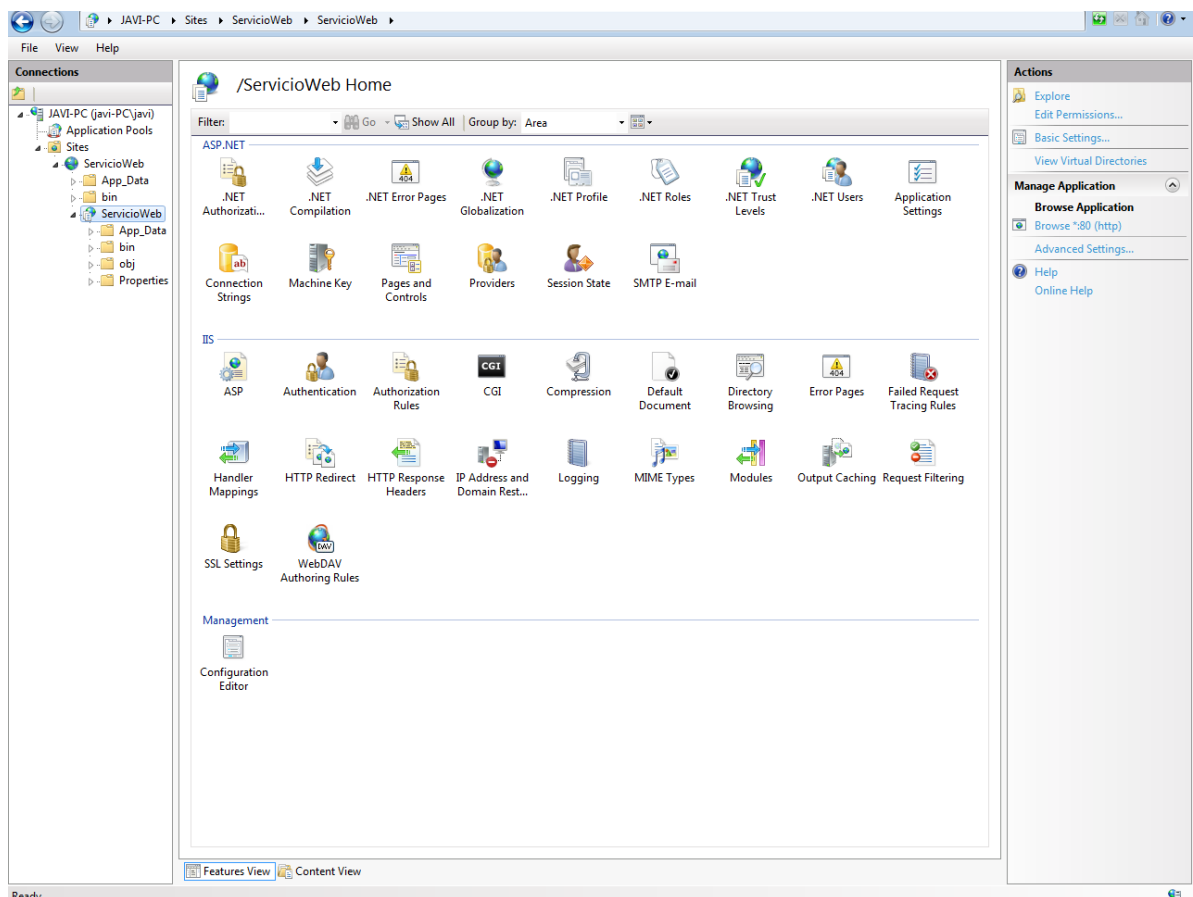


Ilustración 21: Configuración IIS

### 5.1.2. Implementación del servicio web

A continuación se crea el servicio web en el servidor creado en el paso anterior. Para crear el servicio web, se ha implementado en un lenguaje C# debido a que es un lenguaje potente y eficaz que se encuentra como vimos en puntos anteriores, dentro del Framework de ASP .NET, y por lo tanto nos servirá de manera eficaz para implementar posteriormente las comunicaciones entre las aplicaciones cliente y el servidor.

Como se explicó en el capítulo del Estado del Arte, el servicio web funciona como una capa intermedia que enlaza en mi caso cada aplicación Android con una base de datos que se creará posteriormente. De esta manera se hace transparente el acceso a distintos contenidos web por parte de las aplicaciones Cliente-Servidor.

El servicio web creado consta de una clase principal, la cual funciona como controlador del servicio web. El objetivo de esta clase es distribuir de forma eficaz las peticiones realizadas por el cliente entre otras clases que agrupan los distintos métodos en funciones similares. Los métodos de esta clase contienen la etiqueta *[WebMethod]* que indica que son métodos que se usarán fuera del servicio web, es decir, en las aplicaciones Cliente mediante el protocolo HTTP.

El servicio web creado sirve además, para poderse comunicar con la base de datos que se creará posteriormente. A través del servicio web se pueden obtener todos los datos necesarios de la base de datos asegurando la privacidad de la información ya que únicamente se obtienen los datos a través del servicio web, asegurando que no se accede a no ser que sea un usuario con los privilegios suficientes.

A la hora de implementar el servicio web, tal y como se menciona anteriormente, se ha creado la posibilidad de que se envíen correos a los distintos usuarios, siempre que la aplicación lo requiera. Para ello hay que establecer una comunicación con el protocolo SMTP con el puerto 25, permitiendo enviar correos electrónicos. A estas funciones hay que proporcionarles parámetros como los que se muestran a continuación:

- **Destinatario:** usuario al que le llega el correo electrónico, que en el caso del sistema es el usuario final que interacciona con la aplicación cliente.
- **Remitente:** persona que envía el correo electrónico, que en este caso es el servidor.
- **Asunto:** es un breve resumen del contenido.
- **Cuerpo:** contiene el contenido que le va a llegar al usuario.

### 5.1.3. Creación de la base de datos

A continuación se procede a crear la base de datos la cual se alojará lógicamente en el servidor. Para crear esta base de datos, es necesario tener instalado previamente el entorno de programación Visual Studio .NET, tal y como se explicó en el Estado del Arte. La aplicación requiere de una base de datos debido a que es un requisito el poder almacenar y gestionar los libros de la biblioteca, así como la información relativa de los distintos usuarios.

En un principio se planteó almacenar los datos en la aplicación cliente, pero para mantener la seguridad y hacer las aplicaciones cliente más ligeras, se decidió almacenarlos externamente. Los datos que se alojan, son únicamente por el servicio web, de modo que se cumpla el patrón previamente explicado MVC. Esta decisión ha sido tomada para mantener la integridad de los mismos, así como la confidencialidad de los distintos datos de los clientes.

### 5.1.4. Implementación de las pantallas Android

Una vez creadas la base de datos, el servidor y el servicio web, se van a implementar las distintas pantallas de las dos aplicaciones cliente. Esta parte es fundamental en el proyecto, debido a que sin la creación de estas pantallas no es posible que el usuario final interactúe con la aplicación, lo que supondría que no se puede usar. Como los módulos internos de cada aplicación son similares en su estructura, se van a comentar para explicar de una forma más precisa la funcionalidad de dichas pantallas por separado.

A la hora de crear las distintas pantallas de las aplicaciones Cliente, no se va a tener en cuenta la comunicación con el servidor debido a que eso se va a explicar en el siguiente apartado.

Los productos generados al desarrollar las interfaces de usuario en Android, han sido dos, debido a que están dirigidos a distintos usuarios finales del sistema. Los productos generados son los que se explican a continuación.

#### ***Aplicación administrador***

Esta aplicación está destinada a aquellos usuarios finales que sean administradores del sistema, o lo que es lo mismo que sean los bibliotecarios responsables del préstamo y la devolución de los distintos ejemplares.

Esta aplicación tiene de especial, que es posible administrar y controlar al usuario y libros del sistema, desde tal punto que el usuario que controle esta aplicación puede gestionar cuatro tipos de controles de administración básicos tanto para los usuarios, como para los libros. A su vez esta aplicación permite la gestión de préstamos y devoluciones de libros.



Bibliotecario

## Aplicación Gestor

**Usuario**

**Contraseña**

**Aceptar** **Cancelar**

Ilustración 22: Pantalla de ejemplo de la aplicación administrador

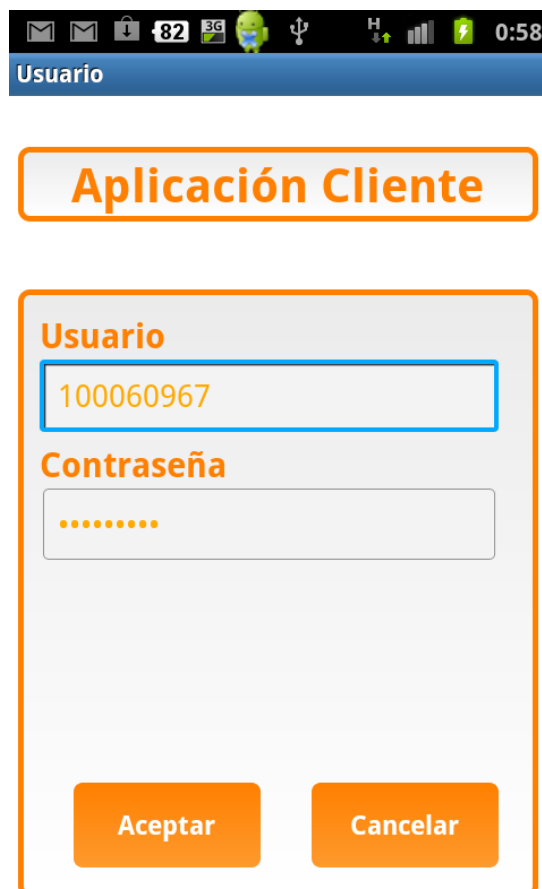
### ***Aplicación usuario***

Esta aplicación está destinada a aquellos usuarios finales que no sean administradores del sistema, o lo que es lo mismo los usuarios a los que se va a terminar prestando libros.

Esta aplicación es más básica que la anterior debido a que no gestiona ningún tipo de usuario ni libro como hace su homóloga. En cambio, esta aplicación permite el control constante de los eventos relacionados con la cuenta del usuario tales como ver su perfil, ver los préstamos actuales y ver un historial de préstamos realizados.

A su vez esta aplicación permite a diferencia de la aplicación administrador, la renovación y reserva de un préstamo. Por último comentar que esta aplicación permite al usuario realizar un autoprestamo obteniendo el libro directamente de la biblioteca sin que llegue a pasar por las manos del bibliotecario.

Estas dos aplicaciones Android se explicarán con un mayor detalle en el *Anexo B* (Manual de usuario) que se adjunta en este documento, con el fin de poder ayudar al usuario en todo lo relativo a las distintas funcionalidades y usos del sistema resultándole mucho más sencillo de usar.



**Ilustración 23: Pantalla de ejemplo de la aplicación usuario**

### 5.1.5. Gestión de las comunicaciones

A continuación se crean las comunicaciones de la aplicación, de tal forma que sea capaz de poder comunicarse entre las aplicaciones cliente y el servidor cumpliendo tal y como se vio anteriormente cuándo se explicó el patrón Cliente-Servidor. Debido a que el sistema al completo creado es un sistema distribuido, es necesario que el cliente tenga que realizar múltiples peticiones al servidor de forma que únicamente pueda obtener los datos si establece comunicación con el.

Cada una de las peticiones creadas en las aplicaciones cliente no son genéricas, es decir, cada petición enviada al servidor tiene su propio código debido a que no siempre se requiere obtener del servidor la misma información. Por este motivo se han creado una serie de funciones que permitan la comunicación con el servidor de forma que se obtengan todos los datos que los usuarios solicitan en los dispositivos móviles. De esta forma se obtiene un control más exhaustivo de cada uno de los errores debido a que se pueden aislar más fácilmente y se evita redundancia a la hora de crear código ya que únicamente se ejecuta la parte de código que sea necesario para esa petición.

A la hora de crear las comunicaciones en primer lugar hay que incluir una librería en cada aplicación cliente que se quiere crear. Esto es debido a que no se incluyen de serie en el código nativo de Android, por lo que es necesario incluir algún tipo de librería que nos permita acceder mediante SOAP al Servicio Web. Por esto se ha obtenido una librería externa llamada "*kSoap2-android*". Esta librería está especialmente adaptada para la tecnología Android, de forma que se permita la comunicación entre cada aplicación Android de una forma fácil y cómoda utilizando servicios web a través del estándar SOAP, y que se nos permita comunicar con el servidor realizándole peticiones HTTP.

A la hora de diseñar las comunicaciones del lado de las aplicaciones Android se han tenido en cuenta, cuatro atributos esenciales que nos garanticen la correcta comunicación entre la aplicación cliente y el servicio web. Dichos atributos son:

- **NAMESPACE:** espacio de nombres utilizado en el servicio web.
- **URL:** dirección URL para realizar la conexión con el servicio web. En el caso de la aplicación se sustituye por la Url donde está instalado el servidor.
- **NOMBRE\_METODO:** nombre del método web concreto que se va a ejecutar. Es el nombre propio del método que se quiere ejecuta en el servicio web.
- **ACCION\_SOAP:** equivalente a NOMBRE\_METODO, pero en la notación definida por SOAP.

Una vez definidos los valores constantes que permitirán la comunicación entre el cliente y el servidor, en primer lugar crearemos la petición. Para ello crearemos un nuevo objeto *SoapObject* pasándole el *Namespace* mencionado anteriormente, y el nombre del método web. A esta petición tendremos que asociar los parámetros de entrada al que pasaremos los nombres y valores de los parámetros.

El segundo paso será crear el envoltorio de SOAP y asociarle la petición deseada por el cliente. Para ello se va a crear un objeto *SoapSerializationEnvelope* indicando la versión de SOAP que vamos a usar que por defecto va a ser la versión 1.1. Indicaremos además que se trata de un servicio web .NET activando su propiedad *dotNet*. Por último, asociaremos la petición antes creada a nuestro contenedor llamando al método *setOutputSoapObject()*.

A continuación se crea el objeto que se encarga de realizar la comunicación HTTP con el servidor, de tipo *HttpTransportSE*, al que pasaremos la URL de conexión a nuestro servicio web. Por último, completaremos el proceso realizando la llamada al servicio web mediante la llamada al método *call()*.

Por último desde la aplicación Android, hay que darle permisos para poder comunicar datos por Internet, ya que Android incluye mecanismos de seguridad y si no es necesario no se permiten las comunicaciones fuera del dispositivo. Por este motivo hay que otorgar privilegios a cada aplicación para ser usado en Internet.

#### 5.1.6. Servicio de correo

Dentro del proyecto .NET creado, se ha creado un proyecto destinado a proporcionar un servicio de correo. Este servicio sirve para poder enviar correos electrónicos a aquellos usuarios que realicen ciertos eventos al interactuar con el sistema. En concreto se envía un correo a los distintos usuarios cada vez que uno de ellos realiza alguna de las siguientes actividades:

- **Préstamo de un libro:** se envía un correo a aquellos usuarios que hallan realizado el préstamo de alguno de los libros que se encuentran registrados en el sistema.
- **Reserva de un libro:** se envía un correo a aquellos usuarios que hallan realizado la reserva de alguno de los libros que se encuentran registrados en el sistema, siempre que este libro no esté previamente prestado.
- **Renovación de un préstamo:** se envía un correo a aquellos usuarios que hallan realizado la renovación del préstamo de alguno de los libros que se tiene previamente prestados.



- **Finalización de un préstamo:** se envía un correo a aquellos usuarios a los que les vaya a caducar el préstamo de un libro en un día, recordándoles que tienen que devolverlo antes de su finalización de préstamo.
- **Finalización de la reserva de un libro:** se envía un correo a aquellos usuarios a los que les vaya a caducar la reserva de un libro en un día. En caso de que finalice el plazo, el libro queda automáticamente libre para su préstamo.
- **Penalización:** se envía un correo a aquellos usuarios que estén penalizados por haber entregado algún ejemplar con retraso.

Cada uno de estos correos se envía debido a que se ha creado un ejecutable que está constantemente activo y una vez al día mediante la programación de la tarea comprueba cada uno de los usuarios a los que se les tiene que enviar un correo electrónico recordándoles alguno de los eventos citados anteriormente.

### 5.1.1. Servicio de pruebas unitarias

Dentro del proyecto .NET creado, se ha creado un proyecto destinado a la creación de pruebas unitarias, con el fin de poder probar de forma exhaustiva el servicio web creado, así como la base de datos. Es un proyecto de *Consola*, ejecutando todo en background y devolviendo si ha habido errores a la hora de realizar alguna de las pruebas.

En este proyecto habrá pruebas para probar todo lo relativo a:

- Inserción, borrado, consulta y modificación de libros y usuarios.
- Probar los distintos módulos tales como la penalización de usuarios, el servicio de correo, la renovación y reserva.
- Probar si se generan adecuadamente los préstamos y devoluciones de libros.

## 5.2. Pruebas

A continuación se muestra un conjunto de pruebas que han sido seleccionadas de un conjunto que se han realizado a la hora de desarrollar el proyecto. En el conjunto de pruebas hay que destacar, que no son las únicas que se han realizado, ya que son numerosas las pruebas que han sido susceptibles de analizar para la comprobación de la integridad del sistema y por lo tanto, por su elevado número, no es viable ponerlas todas en dicho documento. Por este motivo se van a mostrar las principales y más importantes.

Las pruebas han servido para demostrar que todos los requisitos de software se han cumplido. Al final de este apartado se muestra una matriz de trazabilidad que permite asegurar dicha afirmación.

### 5.2.1. Pruebas de aceptación

En este apartado se procederá a definir el plan de pruebas que se va a llevar a cabo en el proyecto para asegurar una calidad, completitud y un correcto funcionamiento del sistema. Para el diseño de las pruebas, se crearán casos de prueba en los cuales se especifica lo que se va a probar y como realizar dicha prueba.

En la siguiente tabla se muestra el modelo que se va a usar para definir las pruebas:

Identificador	PA-XX
Descripción	

Tabla 107: Plantilla de Pruebas de aceptación

- **Identificador:** es el identificador de la prueba. Está formado por una cadena fija “PA” y dos dígitos variables “XX”.
- **Descripción:** se describe de manera breve en que consiste la prueba y el resultado esperado.

Identificador	PA-01
Descripción	Desde la aplicación del usuario, se procede a insertar el usuario y contraseña de un usuario cualquiera de forma que sea posible acceder a dicha aplicación.

Tabla 108: PA-01

Identificador	PA-02
Descripción	Desde la aplicación del administrador, se procede a insertar el usuario y contraseña del administrador, de forma que sea posible acceder a dicha aplicación.

Tabla 109: PA-02

Identificador	PA-03
Descripción	<p>Una vez iniciada sesión de la aplicación del administrador, situarnos en la pestaña de “Info. Usuario”. A continuación se procede a insertar los datos de registro de un nuevo usuario pulsando en el botón “Añadir” introduciendo los datos necesarios.</p> <p>Seguidamente se comprobará desde la aplicación usuario que efectivamente ha sido creado introduciendo desde la ventana de inicio de sesión los datos de “Usuario” y “Contraseña”.</p>

Tabla 110: PA-03

Identificador	PA-04
Descripción	<p>Una vez iniciada sesión de la aplicación del administrador, situarnos en la pestaña de “Info. Usuario”. A continuación se procede a borrar los datos de un usuario pulsando en el botón “Borrar” introduciendo el identificador de usuario que se quiere eliminar.</p> <p>Seguidamente se comprobará desde la aplicación usuario que efectivamente ha sido borrada introduciendo desde la ventana de inicio de sesión los datos de “Usuario” y “Contraseña” y no permitiendo su acceso.</p>

Tabla 111: PA-04

Identificador	PA-05
Descripción	Una vez iniciada sesión en la aplicación cliente, presionar el botón “Consultar” para activar la cámara. Escanear un libro y comprobar que se visualiza correctamente en el dispositivo.

Tabla 112: PA-05

Identificador	PA-06
Descripción	Una vez iniciada sesión en la aplicación administrador, presionar el botón “Consultar libro” para activar la cámara. Escanear un código de barras de un libro y comprobar que se visualiza correctamente en el dispositivo.

Tabla 113: PA-06

Identificador	PA-07
Descripción	Una vez iniciada sesión en la aplicación administrador, presionar el botón “Consultar usuario” para activar la cámara. Escanear un carnet de estudiante de un usuario y comprobar que se visualiza correctamente en el dispositivo.

Tabla 114: PA-07

Identificador	PA-08
Descripción	Una vez iniciada sesión de la aplicación del administrador, situarnos en la pestaña de “Info. Usuario”. A continuación se procede a consultar el usuario pulsando en el botón “Consultar”. En la nueva pantalla elegir el modo que se quiere consultar, ya sea de forma manual introduciendo su identificador de usuario o escaneando el carnet del estudiante. Se debe mostrar en la pantalla los datos del usuario correctamente.

Tabla 115: PA-08

Identificador	PA-09
Descripción	Una vez iniciada sesión de la aplicación del administrador, situarnos en la pestaña de “Info. Usuario”. A continuación se procede a modificar el usuario pulsando en el botón “Modificar”. En la nueva pantalla elegir el modo que se quiere modificar, ya sea de forma manual introduciendo su identificador de usuario o escaneando el carnet del estudiante. A continuación se debe mostrar en la pantalla siguiente los datos del usuario a modificar. Comprobar desde la misma pestaña, pulsando sobre el botón Consultar que efectivamente se han modificado los datos.

Tabla 116: PA-09

Identificador	PA-10
Descripción	Una vez iniciada sesión de la aplicación del administrador, situarnos en la pestaña de “Info. Libro”. A continuación se procede a insertar los datos de registro de un nuevo libro pulsando en el botón “Añadir” introduciendo los datos necesarios. Seguidamente se comprobará desde la aplicación administrador, desde el botón Consultar, que efectivamente ha sido creado.

Tabla 117: PA-10

Identificador	PA-11
Descripción	Una vez iniciada sesión de la aplicación del administrador, situarnos en la pestaña de “Info. Libro”. A continuación se procede a borrar los datos de un libro pulsando en el botón “Borrar” introduciendo el identificador del libro que se quiere eliminar. Seguidamente se comprobará desde la aplicación administrador, desde el botón Consultar, que efectivamente ha sido eliminado.

Tabla 118: PA-11

Identificador	PA-12
Descripción	Una vez iniciada sesión de la aplicación del administrador, situarnos en la pestaña de “Info. Libro”. A continuación se procede a consultar el libro pulsando en el botón “Consultar”. En la nueva pantalla elegir el modo que se quiere consultar, ya sea de forma manual introduciendo su identificador de libro o escaneando el código de barras. Se debe mostrar en la pantalla los datos del libro correctamente.

Tabla 119: PA-12

Identificador	PA-13
Descripción	Una vez iniciada sesión de la aplicación del administrador, situarnos en la pestaña de “Info. Libro”. A continuación se procede a modificar el libro pulsando en el botón “Modificar”. En la nueva pantalla elegir el modo que se quiere modificar, ya sea de forma manual introduciendo su identificador de libro o escaneando el código de barras. A continuación se debe mostrar en la pantalla siguiente los datos del libro a modificar. Comprobar desde la misma pestaña, pulsando sobre el botón Consultar que efectivamente se han modificado los datos.

Tabla 120: PA-13

Identificador	PA-14
Descripción	Una vez iniciada sesión de la aplicación del usuario, se procede a pulsar el botón “Perfil”. En esta pantalla se procede a pulsar el botón “Mi perfil” accediendo al perfil del usuario y visualizándolo correctamente en el dispositivo.

Tabla 121: PA-14

Identificador	PA-15
Descripción	Una vez iniciada sesión de la aplicación del usuario, se procede a pulsar el botón “Perfil”. En esta pantalla se procede a pulsar el botón “Mis préstamos” accediendo a los préstamos actuales del usuario y visualizándolo correctamente en el dispositivo.

Tabla 122: PA-15

Identificador	PA-16
Descripción	Una vez iniciada sesión de la aplicación del usuario, se procede a pulsar el botón “Perfil”. En esta pantalla se procede a pulsar el botón “Mi historial” accediendo al historial de los diez últimos préstamos del usuario y visualizándolo correctamente en el dispositivo.

Tabla 123: PA-16

Identificador	PA-17
Descripción	<p>Una vez iniciada sesión de la aplicación del usuario, se procede a pulsar el botón “Configuración”. En esta pantalla se procede a pulsar el botón “Cambiar contraseña” introduciendo la antigua contraseña, la nueva contraseña y repetir por motivos de seguridad, la nueva contraseña accediendo al historial de los diez últimos préstamos del usuario y visualizándolo correctamente en el dispositivo.</p> <p>Seguidamente se comprobará desde la aplicación usuario que efectivamente se ha cambiado la contraseña introduciendo nueva contraseña y accediendo a la aplicación.</p>

Tabla 124: PA-17

Identificador	PA-18
Descripción	<p>Una vez iniciada sesión de la aplicación del administrador, se procede a pulsar el botón “Configuración”. En esta pantalla se procede a pulsar el botón “Cambiar contraseña” introduciendo la antigua contraseña, la nueva contraseña y repetir por motivos de seguridad, la nueva contraseña accediendo al historial de los diez últimos préstamos del usuario y visualizándolo correctamente en el dispositivo.</p> <p>Seguidamente se comprobará desde la aplicación usuario que efectivamente se ha cambiado la contraseña introduciendo nueva contraseña y accediendo a la aplicación.</p>

Tabla 125: PA-18

Identificador	PA-19
Descripción	<p>Una vez iniciada sesión de la aplicación del usuario, se procede a pulsar el botón “Préstamo”. En la nueva pantalla elegir la forma en que se quiere realizar el préstamo, ya sea de forma manual introduciendo su identificador de libro o escaneando el código de barras.</p> <p>A continuación hacer click en el botón “Aceptar” y se nos mostrará un identificador que habrá que introducir en la máquina que permite el desmagnetizado del libro.</p>

Tabla 126: PA-19

Identificador	PA-20
Descripción	<p>Una vez iniciada sesión de la aplicación del administrador, se procede a pulsar el botón “Préstamo”. En la nueva pantalla elegir la forma en que se quiere realizar el préstamo, ya sea de forma manual introduciendo el identificador de libro o escaneando el código de barras, así como la introducción de los datos del usuario que quiere realizar el préstamo.</p> <p>A continuación hacer click en el botón “Aceptar” y se nos mostrará un identificador que habrá que introducir en la máquina que permite el desmagnetizado del libro.</p>

Tabla 127: PA-20

Identificador	PA-21
Descripción	<p>Una vez iniciada sesión de la aplicación del administrador, se procede a pulsar el botón “Devolución”. En la nueva pantalla elegir la forma en que se quiere realizar el préstamo, ya sea de forma manual introduciendo el identificador de libro o escaneando el código de barras.</p> <p>A continuación hacer click en el botón “Aceptar” y se nos mostrará un identificador que habrá que introducir en la máquina que permite el desmagnetizado del libro.</p>

Tabla 128: PA-21

Identificador	PA-22
Descripción	<p>Una vez iniciada sesión de la aplicación, y tras haber realizado un préstamo, el usuario debe comprobar su correo universitario para ver si se le ha notificado que tiene un préstamo.</p> <p>En dicho correo se le indica que tiene un préstamo nuevo, el título del libro que se ha prestado y la fecha fin del préstamo.</p>

Tabla 129: PA-22

Identificador	PA-23
Descripción	<p>Una vez iniciada sesión de la aplicación, y tras haber realizado un préstamo, y estar penalizado, se comprobará la recepción del correo electrónico</p> <p>En dicho correo se le indica que tiene una penalización.</p>

Tabla 130: PA-23

Identificador	PA-24
Descripción	<p>Una vez iniciada sesión de la aplicación, y tras haber realizado un préstamo, estando a tres días de la caducidad del préstamo, el usuario debe comprobar que puede renovar perfectamente un préstamo.</p> <p>Del mismo modo, el usuario debe comprobar que estando fuera de plazo no puede renovar el préstamo</p>

Tabla 131: PA-25

Identificador	PA-25
Descripción	Una vez iniciada sesión de la aplicación, y sin haber realizado un préstamo, el usuario debe comprobar que puede reservar perfectamente un libro. Comprobar que ese libro a los tres días de no recogerlo, vuelve a estar libre para su préstamo.

Tabla 132: PA-25

### 5.3. Matrices de trazabilidad

En este apartado se van a mostrar diferentes matrices de trazabilidad, para asegurar que se cumplen los requisitos software frente a los requisitos de usuario, así como de las pruebas del sistema frente a los requisitos de usuario.

Las matrices de trazabilidad que se van a mostrar son:

- Matriz de trazabilidad RUC – RSF
- Matriz de trazabilidad RUR – RSNF
- Matriz de trazabilidad PA – RSF

A continuación se puede observar cada una de las matrices descritas anteriormente.

#### 5.3.1. Matriz de trazabilidad RUC - RSF





	RF-01	RF-02	RF-03	RF-04	RF-05	RF-06	RF-07	RF-08	RF-09	RF-10	RF-11	RF-12	RF-13	RF-14	RF-15	RF-16	RF-17	RF-18	RF-19	RF-20	RF-21	RF-22
RUC-01						X				X	X											
RUC-02		X	X	X	X																	
RUC-03	X						X	X	X													
RUC-04									X	X	X											
RUC-05					X																	
RUC-06									X													
RUC-07					X																	
RUC-08		X																				
RUC-09						X																
RUC-10				X																		
RUC-11				X																		
RUC-12								X														
RUC-13								X														
RUC-14			X																			
RUC-15			X																			
RUC-16							X															
RUC-17							X															
RUC-18										X												
RUC-19										X												
RUC-20											X											
RUC-21										X												
RUC-22												X										
RUC-23													X									
RUC-24														X								
RUC-25															X							
RUC-26																X						
RUC-27																	X					
RUC-28																		X				

RUC-29																		X				
RUC-30																			X			
RUC-31																				X		
RUC-32																				X		
RUC-33																				X		
RUC-34																					X	
RUC-35																						X

Tabla 133: Matriz d trazabilidad RC – RSF

### 5.3.2. Matriz de trazabilidad RUR - RSNF

	RSNF-01	RSNF-02	RSNF-03	RSNF-04	RSNF-05	RSNF-06	RSNF-07	RSNF-08	RSNF-09	RSNF-10	RSNF-11	RSNF-12
RUR-01	X											
RUR-02		X										
RUR-03			X									
RUR-04					X							
RUR-05				X								
RUR-06						X	X					
RUR-07									X			
RUR-08										X		
RUR-09											X	
RUR-10											X	
RUR-11												X
RUR-12								X				

Tabla 134: Matriz d trazabilidad RUR – RSNF

### 5.3.3. Matriz de trazabilidad PA - RSF

	RSF-01	RSF-02	RSF-03	RSF-04	RSF-05	RSF-06	RSF-07	RSF-08	RSF-09	RSF-10	RSF-11	RSF-12	RSF-13	RSF-14	RSF-15	RSF-16	RSF-17	RSF-18	RSF-19	RSF-20	RSF-21	RSF-22
PA-01	X																		X			
PA-02	X																					
PA-03		X																				
PA-04				X																		
PA-05					X																	
PA-06									X													
PA-07					X																	
PA-08					X																	
PA-09			X																			
PA-10						X																
PA-11								X														
PA-12									X													
PA-13							X															
PA-14															X							
PA-15														X								
PA-16													X									
PA-17												X										
PA-18												X										
PA-19										X												
PA-20										X						X			X	X		
PA-21											X					X						
PA-22																	X					
PA-23																		X				
PA-24																					X	
PA-25																						X

Tabla 135: Matriz d trazabilidad PA - RSF

Como se puede apreciar en cada una de las matrices, se cumplen todos los requisitos de software propuestos frente a los requisitos de usuario. Se aprecia como cada uno de los requisitos de usuario se corresponde con uno o varios de software por lo que se pueden asegurar que se han implementado cada uno de los requisitos.

En el caso de las pruebas del sistema, se aprecia que se cumplen cada una de las pruebas frente a los requisitos software funcionales. Se debe tener en cuenta que para asegurar que el sistema funciona perfectamente, hay que realizar una mayor batería de pruebas, ya que estas prueban el sistema perfectamente pero pueden realizarse más profundas. Esto sería aconsejable en el caso de que el sistema propuesto en este proyecto fin de carrera se saque al mercado.



## 6. Conclusiones



Al finalizar este proyecto fin de carrera se han alcanzado varias conclusiones tanto positivas como negativas. A su vez en este apartado se muestran las líneas futuras de próximos desarrollos con objetivo de continuar el desarrollo del proyecto fin de carrera introduciendo mejoras a la solución planteada.

## **6.1. Conclusiones**

En este apartado se muestran las conclusiones alcanzadas al desarrollar el proyecto. Dichas conclusiones son las que se muestran a continuación:

- Con el crecimiento de los Smartphones, ha sido muy interesante y necesario el aprendizaje alcanzado sobre como desarrollar aplicaciones para dispositivos móviles, de dónde se pueden extraer ventajas para un futuro, ya que son unas tecnologías que están muy de moda en estos días cuya tendencia para los próximos años está a la alza. Al renovar un sistema tradicional como es la gestión de préstamos de libros, y al combinarlo con las tecnologías móviles, se crea una herramienta potente como la que se ha desarrollado en este proyecto fin de carrera, la cuál hace que sea usada por el usuario debido a que le resultará más cómo que usar el sistema tradicional.
- A la hora de desarrollar el proyecto fin de carrera se han encontrado varios problemas los cuales se describen a continuación.
  - El primero de ellos está basado en la creación de las interfaces de usuario. Este problema se ha dado debido a que ha costado bastante obtener una interfaz (GUI) que sea atractiva al usuario y a la vez funcional, ya que una aplicación por muy buena que sea, si su interfaz de usuario no es atractiva, hará que el usuario final de la aplicación no le entren ganas de usarla, así como la pérdida de tiempo que se generará si el usuario quiere buscar cierta funcionalidad, además de que si se genera desorden en la aplicación, pues tarda demasiado tiempo en encontrar la funcionalidad que busca, o en el peor de los casos, no la termina encontrando.
  - El segundo problema tiene que ver con el aprendizaje de la tecnología Android combinado con el despliegue del servidor IIS y la comunicación Cliente – Servidor, ya que al principio del desarrollo supuso un problema debido a que no siempre se controlaba la comunicación entre ambas partes, así como el difícil control de errores y depuración de las comunicaciones. Esto supuso enfrentarse al problema y obtener soluciones que hicieron que el sistema fuese robusto frente a errores al controlar todos los errores posibles que se pueden contemplar en el sistema.
- Se han puesto en práctica varios protocolos vistos a lo largo de la carrera, como son el sistema distribuido Cliente – Servidor, el cual ha servido para poder establecer las comunicaciones, asignando un rol de cliente o de servidor a cada una de las partes a las cuáles se les ha definido un cierto número de funcionalidades dependiendo de su rol. A su vez, tal y como se menciona en este

documento, se ha establecido el MVC (Modelo Vista Controlador) desde el que se han definido tres módulos en la aplicación, haciendo que el sistema esté claramente definido mediante módulos haciendo el sistema más efectivo, y por lo tanto teniendo un sistema más modular.

- Se han aprendido nuevas tecnologías como son el montaje de servidores y el desarrollo de aplicaciones Android partiendo desde cero, así como la combinación de las distintas tecnologías, haciendo que trabajen conjuntamente, lo que ha finalizado con el alcance de nuevos conocimientos combinados con otros adquiridos a lo largo de la carrera, por lo que se ha alcanzado uno de los objetivos personales como es el desarrollo de un proyecto combinando la investigación de nuevas tecnologías con lo aprendido hasta el día de hoy.

## **6.2. Líneas futuras**

En este apartado se explican las tendencias del proyecto fin de carrera realizado con objetivo de que se pueda continuar en un futuro mejorando la funcionalidad explicada, destacando una serie de aspectos a mejorar con objeto de una futura implementación.

Uno de los principales objetivos del sistema creado es realizar el préstamo y devolución de libros a distintos usuarios. Debido al incremento de Smartphones en el mercado, y que cada vez son más numerosos el número de usuarios que tienen uno, se ve necesario que en un futuro no muy lejano los usuarios sean capaces de realizar préstamos de una forma más cómoda y sencilla a través de la solución propuesta en este documento.

Para completar este proyecto, se puede realizar una página web que esté en constante armonía con el Servicio Web de forma que cualquier tipo de usuario sea capaz de poder obtener los datos que solicita en tiempo real, y ver cuál es el estado de sus préstamos, así como si quieren realizar una reserva y poder ver la disponibilidad del mismo en la biblioteca.

Otra línea futura que podría ampliar el proyecto sería la de utilizar sistemas de georeferenciación y sistemas GPS. De esta forma sería posible georeferenciar los libros dentro de la biblioteca de forma que el usuario sea capaz de visualizar en su dispositivo móvil, la situación exacta de un libro obteniendo así una facilidad y comodidad de un usuario para poder coger un libro ahorrándose largos tiempos de búsqueda del ejemplar interesado, por el usuario que desea obtener el préstamo.

A su vez, usando tecnologías GPS sería posible obtener la localización exacta de un usuario por parte de la biblioteca. En caso de que no se devolviese un ejemplar en un periodo excesivo de tiempo, de forma que sea fácil para un bibliotecario saber si el usuario ha estado por el entorno de la biblioteca y este no ha devuelto el libro.



Como las redes sociales están tan de moda, tal como es el caso de Twitter y Facebook, sería posible que un usuario a la hora de realizar un préstamo publicase una entrada en la red social que quisiera diciendo que se le ha prestado un libro y lo comentase de forma que sus seguidores se enterasen de la noticia y el contenido del libro.

Como esta aplicación es de muy genérica debido a que es un prototipo, podría ser perfectamente adaptable a sistemas con una base de datos mayor como es el caso de la biblioteca Carlos III de Madrid. Para ello, como la aplicación está realizada mediante módulos, habría que migrar los datos de la biblioteca hacia la base de datos del servidor de forma que estos puedan ser utilizables en todo momento por los administradores de la aplicación o consultados por cada usuario.

Por último comentar que la solución propuesta para poder realizar el autoprestamo, solo sería viable si se modernizara el sistema de desmagnetizado de los libros de la biblioteca que consistiría como ya se ha explicado en dicho documento, en la obtención de una cadena de caracteres a la hora de realizar el préstamo. Dicha cadena de caracteres sería una función Hash del ISBN del libro que la máquina reconocería activando el desmagnetizado del libro, de forma que el usuario se pueda auto prestar el libro, sin tener que recurrir al bibliotecario.





## 7. Acrónimos y abreviaturas



En este apartado se definen varios acrónimos que se encuentran en este documento:

- API            Application Programming Interface
- GPS            Global Position System
- HTTP          HyperText Transfer Protocol
- IDE            Integrated Developed Environment
- JSON          JavaScript Object Notation
- SOAP          Simple Object Access Protocol
- MVC            Model-View-Controller
- UC3M          Universidad Carlos Tercero de Madrid
- URL            Uniform Resource Locator
- XML            Extensible Markup Language
- App            Aplicación móvil
- GUI            Graphic User Interface
- JVM            Java Virtual Machine
- UML            Unified Modeling Language
- IIS            Internet Information Server
- SQL            Structured Query Language
- LINQ           Language Integrated Query
- CLR            Common Language Runtime
- ESA            European Spacial Agent





## 8. Referencias





- [1] Android Wikipedia. 2012. Disponible [Internet]:  
*<http://es.wikipedia.org/wiki/Android>*
- [2] Curso desarrollo de Android. 2012. Disponible [Internet]:  
*<http://www.android.com/>*
- [3] Google Developers. 2012. Disponible [Internet]:  
*<https://developers.google.com/android/>*
- [4] JSON Wikipedia. 2012. Disponible [Internet]:  
*<http://es.wikipedia.org/wiki/JSON>*
- [5] Introducing JSON. 2012. Disponible [Internet]:  
*<http://www.json.org/>*
- [6] Tutorial JSON. 2012. Disponible [Internet]:  
*<http://www.w3schools.com/json/default.asp>*
- [7] SOAP Wikipedia. 2012. Disponible [Internet]:  
*[http://es.wikipedia.org/wiki/Simple\\_Object\\_Access\\_Protocol](http://es.wikipedia.org/wiki/Simple_Object_Access_Protocol)*
- [8] Tutorial SOAP w3schools. 2012. Disponible [Internet]:  
*<http://www.w3schools.com/soap/default.asp>*
- [9] SOAP w3c. 2012. Disponible [Internet]:  
*<http://www.w3.org/TR/soap/>*
- [10] SOAP Tutorial. 2012. Disponible [Internet]:  
*<http://www.desarrolloweb.com/articulos/1557.php>*
- [11] SOAP ksoap2-android. 2012. Disponible [Internet]:  
*<http://code.google.com/p/ksoap2-android/>*
- [12] MVC Wikipedia. 2012. Disponible [Internet]:  
*[http://es.wikipedia.org/wiki/Modelo\\_Vista\\_Controlador](http://es.wikipedia.org/wiki/Modelo_Vista_Controlador)*
- [13] MVC .NET. 2012. Disponible [Internet]:  
*<http://www.asp.net/mvc>*

- [14] .NET Wikipedia. 2012. Disponible [Internet]:  
*[http://es.wikipedia.org/wiki/Microsoft\\_.NET](http://es.wikipedia.org/wiki/Microsoft_.NET)*
- [15] Expósito Singh, David. Curso de Desarrollo de Aplicaciones Distribuidas. Grupo ARCOS, Universidad Carlos III Madrid, 2010.
- [16] C# .NET Wikipedia. 2012. Disponible [Internet]:  
*[http://es.wikipedia.org/wiki/C\\_Sharp](http://es.wikipedia.org/wiki/C_Sharp)*
- [17] Microsoft .NET. 2012. Disponible [Internet]:  
*<http://www.microsoft.com/net>*
- [18] ASP .NET. 2012. Disponible [Internet]:  
*<http://www.asp.net/>*
- [19] SQL Server Express Wikipedia. 2012. Disponible [Internet]:  
*[http://en.wikipedia.org/wiki/SQL\\_Server\\_Express](http://en.wikipedia.org/wiki/SQL_Server_Express)*
- [20] SQL Server Microsoft. 2012. Disponible [Internet]:  
*<http://www.microsoft.com/sqlserver/en/us/editions/2012-editions/express.aspx>*
- [21] SQL Server Tutorial. 2012. Disponible [Internet]:  
*<http://msdn.microsoft.com/es-es/express/aa718378.aspx>*
- [22] IIS Wikipedia. 2012. Disponible [Internet]:  
*[http://es.wikipedia.org/wiki/Internet\\_Information\\_Services](http://es.wikipedia.org/wiki/Internet_Information_Services)*
- [23] IIS Site oficial. 2012. Disponible [Internet]:  
*<http://www.iis.net/>*
- [24] IIS Configuración servicio web. 2012. Disponible [Internet]:  
*<http://technet.microsoft.com/es-es/library/cc754437%28v=ws.10%29.aspx>*
- [25] Linq Wikipedia. 2012. Disponible [Internet]:  
*[http://es.wikipedia.org/wiki/Language\\_Integrated\\_Query](http://es.wikipedia.org/wiki/Language_Integrated_Query)*
- [26] Linq Microsoft. 2012. Disponible [Internet]:  
*<http://msdn.microsoft.com/es-es/library/bb397926.aspx>*

# Anexo A: Planificación y presupuesto



En este apartado, se incluye una planificación detallada del proyecto, así como el presupuesto que ha costado realizar todo el proyecto.

## 1. Planificación

A continuación se muestran los elementos que han sido necesarios para la planificación de este proyecto. Dichos elementos son:

- Coste de personal.
- Coste de hardware y software.

### 1.1 Planificación de tareas

En este apartado se va a especificar la planificación llevada a cabo durante el desarrollo del proyecto, mostrando una lista de las tareas llevadas a cabo, así como un diagrama de Gantt.

El proyecto fin de carrera ha sido realizado en un tiempo alrededor de 5 meses, diferenciándolo en distintos módulos. A continuación se muestran las tareas realizadas junto con el tiempo llevado a cabo y el periodo en el que se han comprendido.

	Nombre de tarea	Duración	Comienzo	Fin
1	<b>Sistema de Control de préstamos</b>	<b>101 días</b>	<b>lun 06/02/12</b>	<b>lun 25/06/12</b>
2	Propuesta del proyecto	6 días	lun 06/02/12	lun 13/02/12
3	<b>Análisis del sistema</b>	<b>16 días</b>	<b>lun 13/02/12</b>	<b>lun 05/03/12</b>
4	Definición del problema	3 días	lun 13/02/12	mié 15/02/12
5	Requisitos de usuario	6 días	jue 16/02/12	jue 23/02/12
6	Casos de uso	3 días	vie 24/02/12	mar 28/02/12
7	Requisitos software	6 días	lun 27/02/12	lun 05/03/12
8	<b>Diseño del sistema</b>	<b>11 días</b>	<b>lun 05/03/12</b>	<b>lun 19/03/12</b>
9	Diseño de la arquitectura	6 días	lun 05/03/12	lun 12/03/12
10	Diseño del sistema	6 días	lun 12/03/12	lun 19/03/12
11	<b>Implementación</b>	<b>35 días</b>	<b>lun 19/03/12</b>	<b>vie 04/05/12</b>
12	Creación del servidor	5 días	lun 19/03/12	vie 23/03/12
13	Creación del servicio web	11 días	lun 26/03/12	lun 09/04/12
14	Creación de la base de datos	6 días	mar 10/04/12	mar 17/04/12
15	Aplicación cliente	8 días	mié 18/04/12	vie 27/04/12
16	Aplicación administrador	6 días	vie 27/04/12	vie 04/05/12
17	<b>Pruebas del sistema</b>	<b>11 días</b>	<b>vie 04/05/12</b>	<b>vie 18/05/12</b>
18	Pruebas de aceptación	6 días	vie 04/05/12	vie 11/05/12
19	Pruebas unitarias	6 días	vie 11/05/12	vie 18/05/12
20	Documentación	31 días	lun 14/05/12	lun 25/06/12

Ilustración 24: Descripción de tareas junto a duración

## 1.2 Diagrama de Gantt.

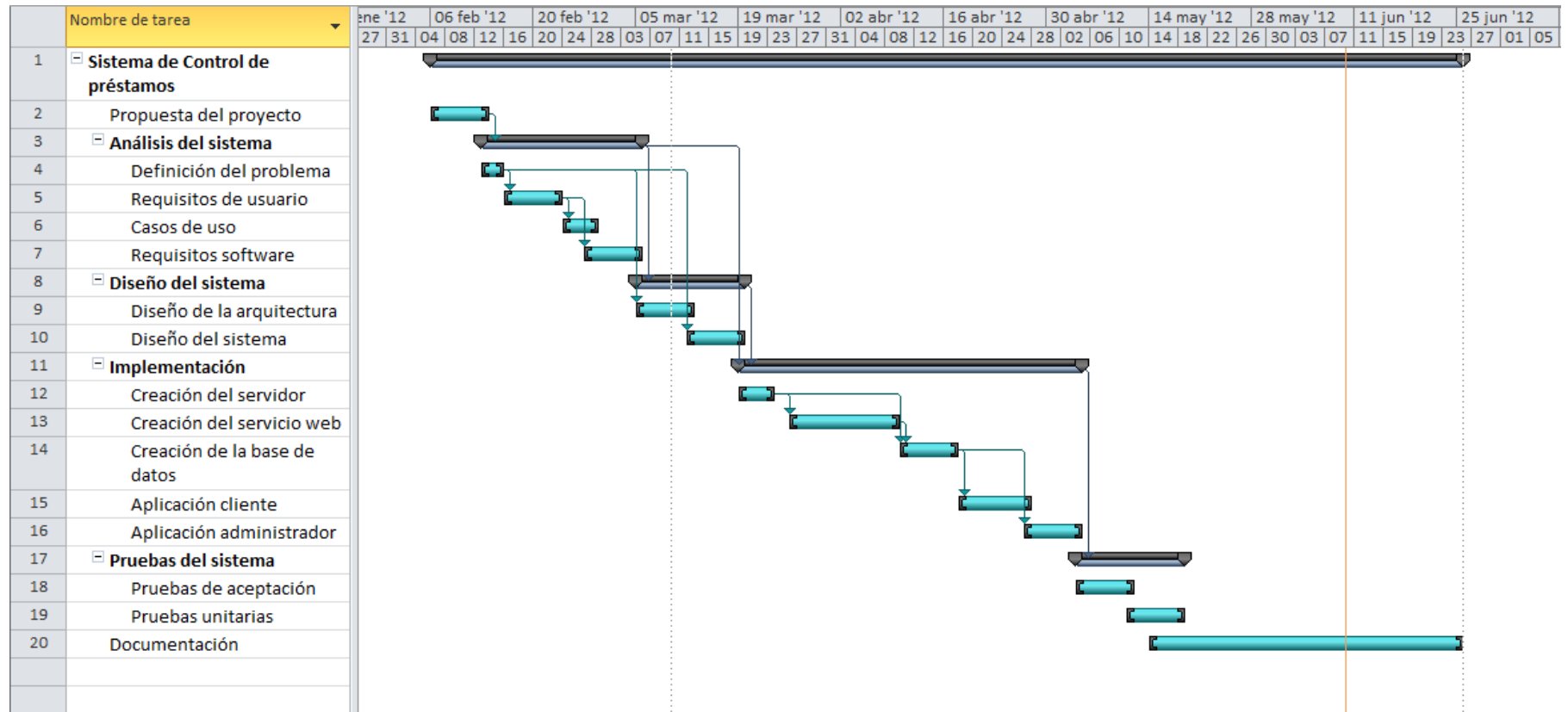


Ilustración 25: Diagrama de Gantt.

## 2. Presupuesto

A continuación se explica un presupuesto detallado del proyecto fin de carrera desde su inicio hasta su finalización.

### 2.1 Costes de personal

Para poder evaluar los costes de personal, ha sido necesario evaluar el tiempo dedicado al desarrollo del proyecto. Para ello, es necesario evaluar diversos aspectos tales como la fecha de inicio y fin del mismo, y el número de horas invertidas.

En la siguiente tabla se muestra el personal que se ha encargado de desarrollar el proyecto. Se ha asumido un coste de 12€/hora teniendo en cuenta que se ha trabajado una media de 8 horas diarias durante 101 días o lo que es lo mismo 4 meses.

Concepto	Coste por día (€)	Número de meses	Coste (€)
<b>Ingeniero Junior</b>	96	101	9696
<b>Total</b>	<b>96</b>	<b>101</b>	<b>9696</b>

Tabla 136: Tabla de coste de personal.

Por lo tanto el coste de personal ha sido de **9696€**.

### 2.2 Costes de Hardware y Software

Para poder evaluar los costes de hardware y software, ha sido necesario evaluar cada uno de los elementos hardware que han intervenido así como el software necesario para el desarrollo del proyecto.

En la siguiente tabla se muestra el personal que se ha encargado de desarrollar el proyecto. Se ha contabilizado todos los elementos necesarios de hardware y software que han se han empleado en el desarrollo del proyecto. Los elemento con (\*) tienen un coste de 0 € debido a que se han obtenido a través de las licencias del MSDN que se conceden por pertenecer a la comunidad universitaria.

Concepto	Cantidad	Coste (€)	Coste (€)
Dispositivo Samsung Galaxy S	1	269	269
HP Pavilion p6-2100	1	699	699
Microsoft Office 2010	1	120	120
Microsoft Project 2010	1	0*	0
Eclipse	1	0*	0
Visual Studio 2010	1	0*	0
<b>Total</b>	<b>6</b>	<b>1088</b>	<b>1088</b>

Tabla 137: Tabla de coste de hardware/software.

Por lo tanto el coste del hardware y del software ha sido de **1088 €**.

### 2.3 Costes Total

A continuación se muestra una tabla en la que se aprecian todos los costes parciales. Se puede apreciar que se muestra el precio con y sin IVA así como un 10% de costes indirectos.

Concepto	Coste (€)
Coste de personal	9696
Coste Hardware/Software	1088
<b>Total</b>	<b>10784</b>
Costes indirectos 10%	1078,4
<b>Coste Final</b>	<b>11863,2</b>

Tabla 138: Tabla de coste de hardware/software.

Por lo tanto el coste final sin IVA es de **11.863,20 € (ONCE MIL OCHOCIENTOS SESENTA Y TRES CON VIENTE CÉNTIMOS)**.

Su precio con IVA del 18% es de **13998,57 €, (TRECE MIL NOVECIENTOS NOVENTA Y OCHO CON CINCUENTA Y SIETE CÉNTIMOS)**.





# Anexo B: Manual de usuario



En este anexo se va a detallar un manual de usuario en el que se va a explicar el funcionamiento de cada una de las aplicaciones Android generadas

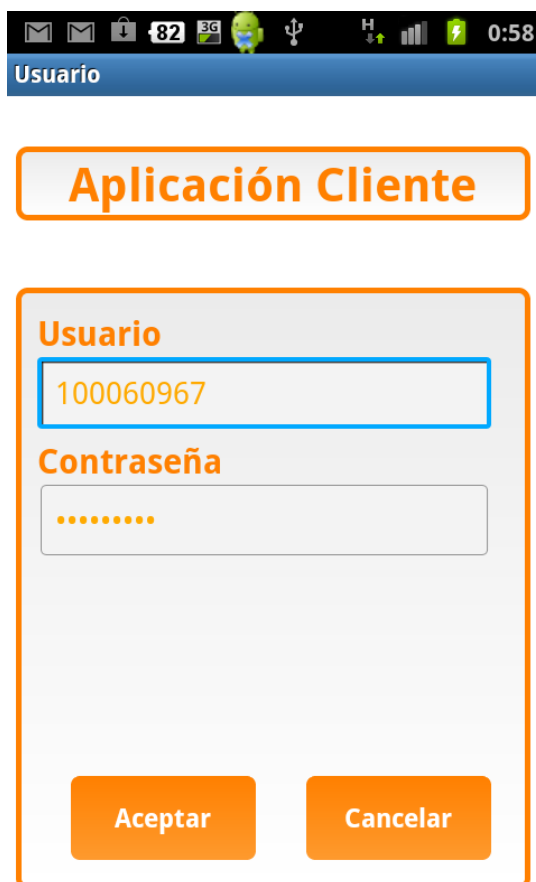
## 1. Aplicación Usuario

Esta aplicación está destinada a aquellos usuarios finales del sistema que no se vayan a dedicar a administrar y gestionar el sistema de control de préstamos.

Las funciones principales de esta aplicación son la información de la actividad y la cuenta del usuario de forma que puede ver su información personal, así como si tiene préstamos o un historial de los libros que ha cogido. Otra función importante es la de poder gestionar sus propios préstamos de forma que sin pasar por el bibliotecario, un usuario es capaz de autoprestarse sus propios libros.

### 1.1 Pantalla de inicio de sesión

Esta pantalla se encuentra tanto en la aplicación administrador como en la aplicación destinada al resto de los usuarios finales. Es la primera pantalla que nos encontramos a la hora de ejecutar cualquiera de las dos aplicaciones. Tiene la funcionalidad de permitir a los usuarios acceder al sistema al introducir sus datos *Usuario* y *Contraseña*.



The image shows a mobile application interface for login. At the top, there is a status bar with various icons and the time 0:58. Below it, a blue header bar contains the word 'Usuario'. The main content area is titled 'Aplicación Cliente' in orange. It features two input fields: 'Usuario' with the value '100060967' and 'Contraseña' with masked characters. At the bottom, there are two orange buttons labeled 'Aceptar' and 'Cancelar'.

Ilustración 26: Pantalla de inicio de sesión.

Esta pantalla permite almacenar los datos de los usuario con el fin de que no sea necesario introducir los datos de inicio de sesión cada vez que el usuario interactúe con la aplicación. Para permitir almacenar las credenciales, se ha hecho uso de un fichero codificado en XML. Cada vez que se quiera hacer uso de los datos del usuario, se procederá a la lectura de este fichero que únicamente se puede acceder a través de la aplicación o a través del usuario Root del dispositivo. A la hora de cerrar la aplicación el fichero se borrará.

## 1.2 Pantalla de menú principal

A esta pantalla se puede acceder únicamente a través de la pantalla de inicio de sesión comentada en el punto anterior. Esta pantalla es la principal de cualquiera de las dos aplicaciones y es necesaria para poder acceder a las distintas funcionalidades de cada aplicación.



Ilustración 27: Pantalla de menú

Estas pantallas permiten entre otras cosas acceder a la configuración del usuario, acceder a la información de la cuenta del mismo, así como poder consultar libros con el escaneador de códigos de barras de la aplicación. En el caso de la aplicación del administrador, se puede añadir, eliminar, consultar y modificar tanto a los libros de la aplicación, así como de los usuarios, permitiendo su gestión.

### 1.3 Pantalla de información de usuario

Esta pantalla se encuentra únicamente en la aplicación del usuario. Sirve para poder gestionar la cuenta del usuario de forma que este pueda ver su información personal, así como los préstamos que tiene actualmente y un historial de los diez últimos préstamos.

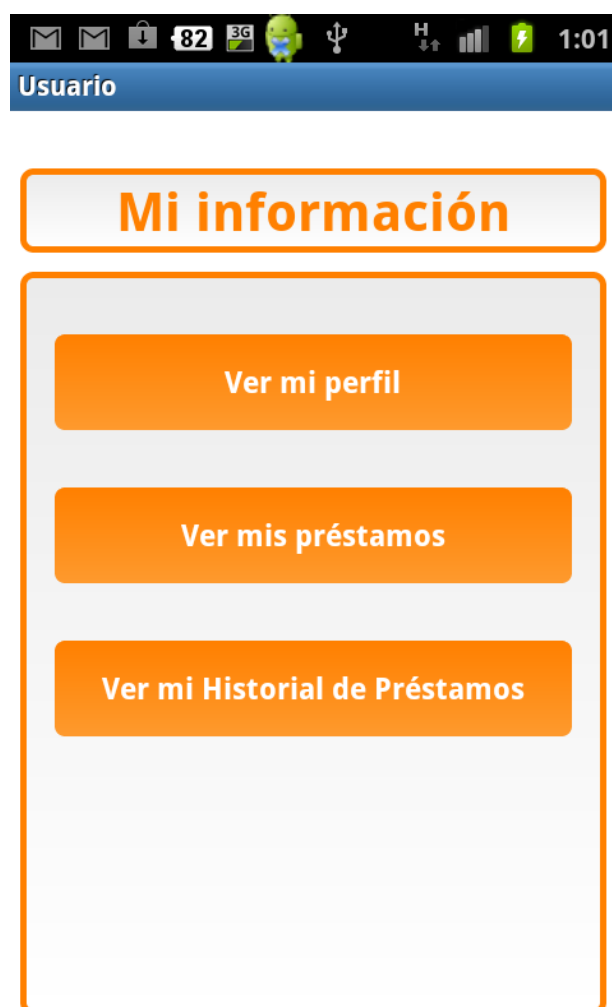


Ilustración 28: Pantalla de información de usuario.

En esta pantalla como se mencionaba anteriormente se pueden ejecutar tres tipos de funciones:

- **Ver información:** permite obtener la información de la cuenta del usuario de forma que pueda ver de forma precisa toda la información de su cuenta. Además se le informa de si tiene préstamos pendientes y de si está penalizado en caso de que lo estuviese.

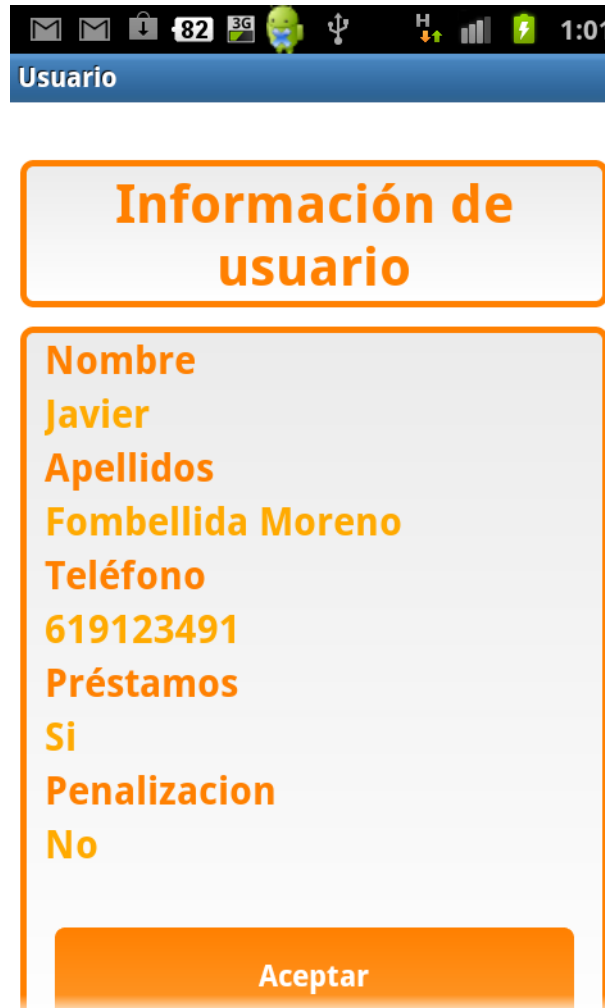


Ilustración 29: Pantalla ver información

- **Ver préstamos:** se muestran los préstamos que tiene actualmente un usuario. En caso de ser un usuario con el rol de alumno, se prestará un máximo de dos libros y en caso de que sea profesor un máximo de cuatro. Además, te informa de la caducidad del préstamo.



Ilustración 30: Pantalla ver préstamos

- **Ver historial de préstamos:** se muestran los diez últimos libros, los cuales han sido prestados al usuario.

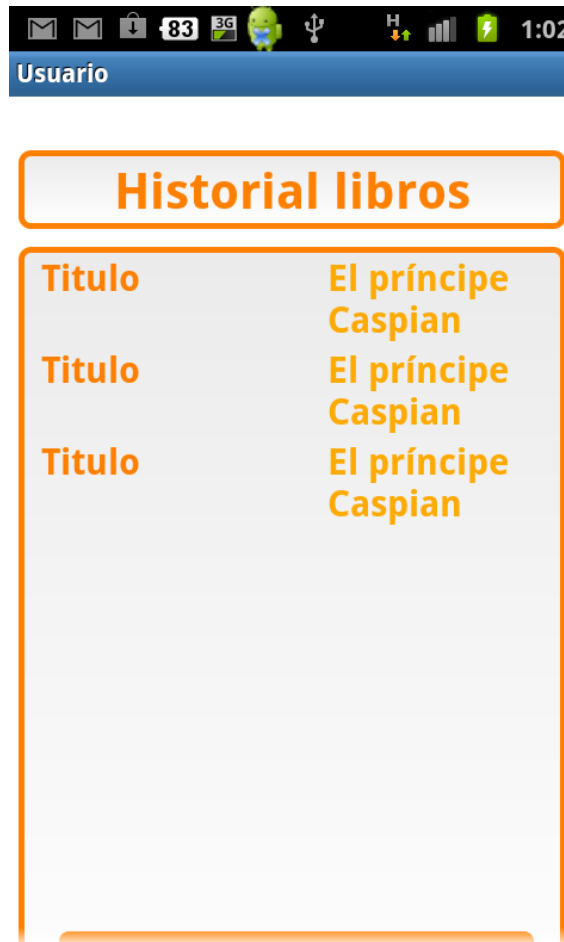


Ilustración 31: Pantalla ver historial



## 1.4 Pantalla de Otros servicios

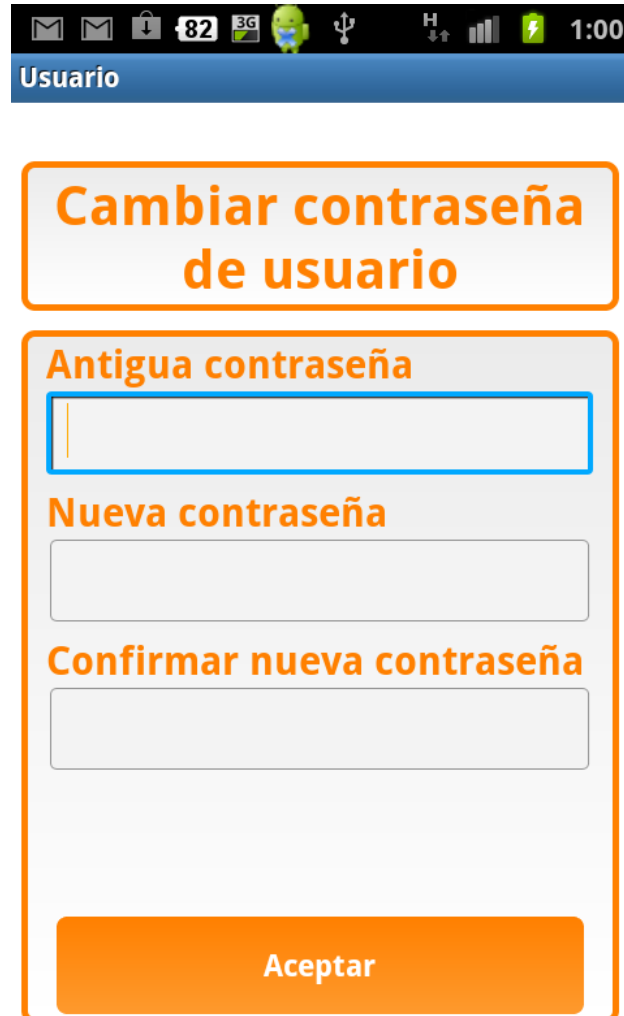
Esta pantalla se encuentra únicamente en la aplicación del usuario. Sirve para poder configurar la cuenta del usuario de forma que este pueda cambiar su contraseña, así como cambiar su cuenta desde la que se permitirá enviar mensajes a las redes sociales.



**Ilustración 32: Pantalla de otros servicios.**

En esta pantalla como se mencionaba anteriormente se pueden ejecutar dos tipos de funciones:

- **Cambiar contraseña:** permite cambiar la contraseña del usuario que inicio sesión. Para poder realizar un cambio de contraseña de forma correcta, es necesario que el usuario introduzca en tres campos los valores de:
  - Contraseña antigua.
  - Contraseña nueva.
  - Repetir contraseña nueva.



Usuario

### Cambiar contraseña de usuario

Antigua contraseña

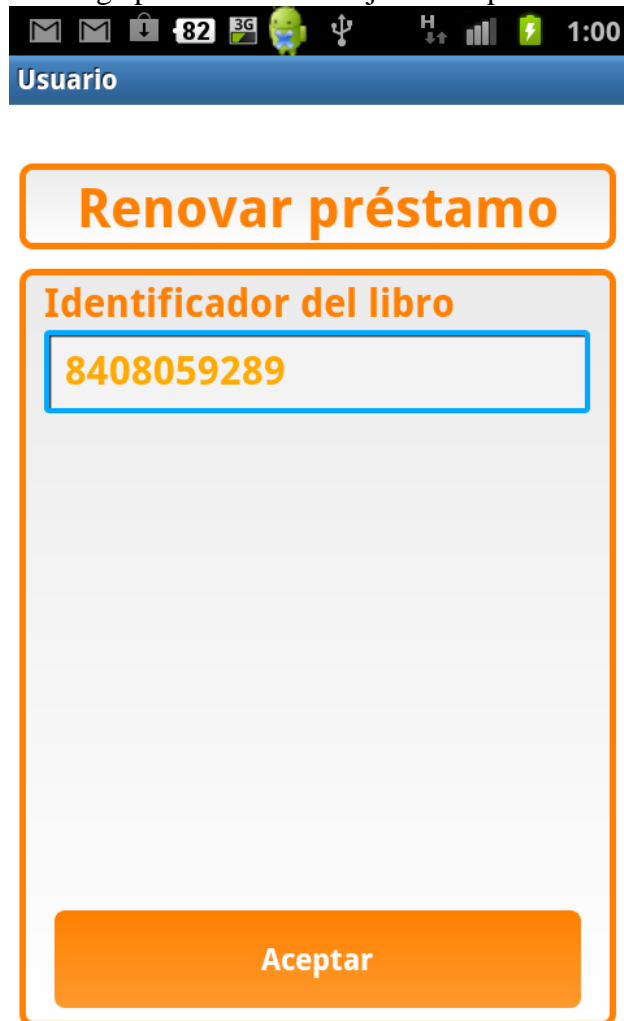
Nueva contraseña

Confirmar nueva contraseña

Aceptar

Ilustración 33: Pantalla de otros servicios.


- **Renovar:** se muestran un menú desde el cual se pide introducir el ISBN de un libro que el usuario tenga prestado con el objetivo de poder renovarlo.



The screenshot shows a mobile application interface. At the top is an Android status bar with icons for email, lock, battery (82%), 3G, a person icon, USB, HSPA+, signal strength, and battery level, along with the time 1:00. Below the status bar is a blue header with the text "Usuario". The main content area has an orange border and contains a title "Renovar préstamo" in orange text. Below the title is a section titled "Identificador del libro" in orange text, which contains a text input field with the value "8408059289". At the bottom of the section is an orange button with the text "Aceptar" in white.

Ilustración 34: Pantalla renovar

- **Reservar:** se muestran un menú desde el cual se pide introducir el ISBN de un libro que se quiera reservar y no esté prestado a ningún usuario.



Usuario

**Reservar**

**Identificador del libro**

8408059289

**Aceptar**

Ilustración 35: Pantalla reservar

## 2. Aplicación administrador

Esta aplicación está destinada a aquellos usuarios finales del sistema que se dedique a administrar y gestionar el sistema de control de préstamos de forma que tengan control sobre los libros que hay en la biblioteca, así como los prestados a los diferentes usuarios.

Además esta aplicación permitirá gestionar los préstamos y devoluciones a los distintos usuarios del sistema de forma que un usuario pueda solicitar un préstamo de un libro o una devolución.

### 2.1 Pantalla de inicio de sesión

Esta pantalla se encuentra tanto en la aplicación administrador como en la aplicación destinada al resto de los usuarios finales. Es la primera pantalla que nos encontramos a la hora de ejecutar cualquiera de las dos aplicaciones. Tiene la funcionalidad de permitir a los usuarios acceder al sistema al introducir sus datos *Usuario* y *Contraseña*.



The image shows a mobile application interface for a library system. At the top, there is a status bar with various icons (lock, mail, 70% battery, 3G, Android robot, USB) and the time 0:08. Below the status bar is a blue header bar with the text 'Bibliotecario'. The main content area has a light gray background. At the top of this area is a blue button labeled 'Aplicación Gestor'. Below this is a login form with a light gray background and a blue border. The form has two input fields: 'Usuario' with the text 'admin' and 'Contraseña' with a masked password represented by dots. At the bottom of the form are two blue buttons: 'Aceptar' and 'Cancelar'.

Ilustración 36: Pantalla de inicio de sesión.

Esta pantalla permite almacenar los datos de los usuario con el fin de que no sea necesario introducir los datos de inicio de sesión cada vez que el usuario interactúe con la aplicación. Para permitir almacenar las credenciales, se ha hecho uso de un fichero codificado en XML. Cada vez que se quiera hacer uso de los datos del usuario, se procederá a la lectura de este fichero que únicamente se puede acceder o por la aplicación o por el usuario Root del dispositivo. A la hora de cerrar la aplicación el fichero se borrará.

## 2.2 Pantallas de Escanear

Esta pantalla se encuentra únicamente en la aplicación del administrador, la cual se puede acceder desde el menú principal. Permite la gestión de escaneo de usuarios y libros para obtener su información de una forma más rápida, así como el control de préstamos y devoluciones.



Ilustración 37: Pantalla de Escanear.

Entre las funciones que contiene, se pueden destacar:

- **Escanear Libro:** permite al usuario administrador escanear el ISBN de un libro obteniendo su información.
- **Escanear Usuario:** permite al usuario administrador escanear el carnet de estudiante de un usuario obteniendo su información.
- **Prestar:** permite al administrador gestionar las devoluciones de libros ya sea de forma manual o mediante el escaneo del carnet de estudiante e ISBN.
- **Devolver:** permite al administrador gestionar las devoluciones de libros ya sea de forma manual o mediante el escaneo del carnet de estudiante e ISBN.

A la hora de prestar o devolver un libro, la aplicación te permite seleccionar de que forma quieres que se realice el préstamo o la devolución, si de forma manual en la que hay que introducir el identificador del libro o del usuario, o de forma de escaneo en la cuál hay que introducir de forma manual cada uno de los identificadores tal y como se pueden ver en las siguientes figuras.



Ilustración 38: Pantalla de selección de modo de préstamo



Ilustración 39: Pantalla de selección de modo de préstamo - escáner



## 2.3 Pantallas de gestión del usuario

Esta pantalla se encuentra únicamente en la aplicación del administrador, la cual se puede acceder desde el menú principal. Permite la gestión de los usuarios que tendrá permitido acceder a la biblioteca.



Ilustración 40: Pantalla de gestión del usuario.

Entre las funciones que contiene, se pueden destacar:

- **Añadir usuario:** permite al usuario administrador, añadir un usuario a la base de datos introduciendo para ello sus datos personales. Si se ha añadido correctamente se indica en la pantalla.
- **Borrar usuario:** permite al usuario administrador, eliminar un usuario de la base de datos introduciendo para ello su identificador único. Si se ha eliminado correctamente se indica en la pantalla.
- **Consultar usuario:** permite al usuario administrador, consultar la información de un usuario a partir de su identificador único. Se muestra toda la información relativa a dicho usuario.

- **Modificar usuario:** permite al administrador modificar la cuenta del usuario que se especifique mediante la introducción de su identificador único.

Para cada una de las cuatro opciones anteriores, la aplicación pide introducir unos parámetros tal y como puede verse en la siguiente figura a modo de ejemplo, en la cuál se muestra la pantalla que aparece al pulsar el botón añadir usuario.



The screenshot shows a mobile application interface for adding a user. At the top, there is a status bar with various icons and the time 0:27. Below it is a blue header bar with the text 'Bibliotecario'. The main content area is a light gray box with a blue border. It contains a title 'Añadir usuario' in blue. Below the title are four input fields, each with a blue border and a blue label above it: 'NIA' with the value '100060967', 'Nombre' with the value 'Javier', 'Apellidos' with the value 'Fombellida Moreno', and 'Teléfono' with the value '619614714'. At the bottom of the form is a blue button with the text 'Aceptar'.

Ilustración 41: Pantalla Añadir un usuario.

## 2.4 Pantallas de gestión del libro

Esta pantalla se encuentra únicamente en la aplicación del administrador, la cual se puede acceder desde el menú principal. Permite la gestión de los libros de la biblioteca.

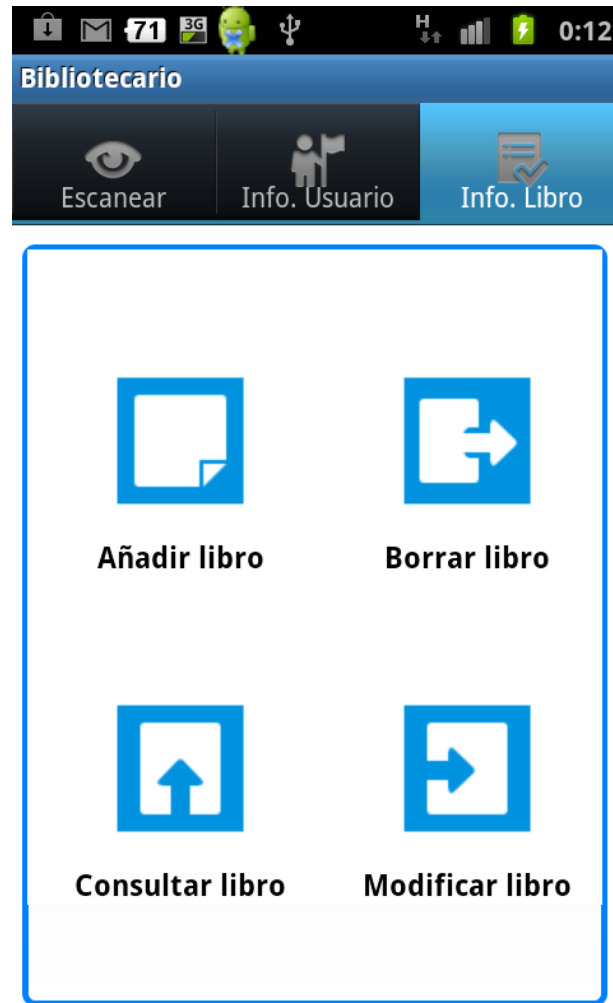


Ilustración 42: Pantalla de gestión del libro.

Entre las funciones que contiene, se pueden destacar:

- **Añadir libro:** permite al usuario administrador, añadir un libro a la base de datos introduciendo para ello sus datos. Si se ha añadido correctamente se indica en la pantalla.
- **Borrar libro:** permite al usuario administrador, eliminar un libro de la base de datos introduciendo para ello su identificador único. Si se ha eliminado correctamente se indica en la pantalla.
- **Consultar libro:** permite al usuario administrador, consultar la información de un libro a partir de su identificador único. Se muestra toda la información relativa a dicho libro.
- **Modificar libro:** permite al administrador modificar el libro que se especifique mediante la introducción de su identificador único.

Para cada una de las cuatro opciones anteriores, la aplicación pide introducir unos parámetros tal y como puede verse en la siguiente figura a modo de ejemplo, en la cuál se muestra la pantalla que aparece al pulsar el botón consultar libro. Se aprecia que nos pide introducir el ISBN o el identificador general de un libro.



The screenshot shows a mobile application interface. At the top is a status bar with various icons and the time 0:29. Below it is a blue header bar with the text 'Bibliotecario'. The main content area has a light gray background. At the top of this area is a blue button with the text 'Consultar libro'. Below the button is a text input field with the label 'ISBN / Id. Libro' and the value '8408059289'. At the bottom of the input field is a blue button with the text 'Aceptar'.

Ilustración 43: Pantalla de consultar libro



# Anexo C: Manual de Implantación



En este anexo se hace un recorrido por cada uno de los módulos que se deben desplegar para poder instalar y configurar el sistema de una forma óptima, para que cada uno de los sistemas interactúen de forma correcta entre sí actuando como una única unidad.

## 1. Aplicaciones Android

Cada una de las aplicaciones Android generadas tienen una extensión .apk. Este tipo de ficheros sirven para instalar aplicaciones en un dispositivo Android, debido a que es el formato que se maneja en este tipo de dispositivos.

- Lo primero de todo hay que localizar el fichero a instalar.
- Una vez localizado el fichero a instalar, hay que activar en el dispositivo lo siguiente:

**Ajustes > Aplicaciones > Orígenes desconocidos**

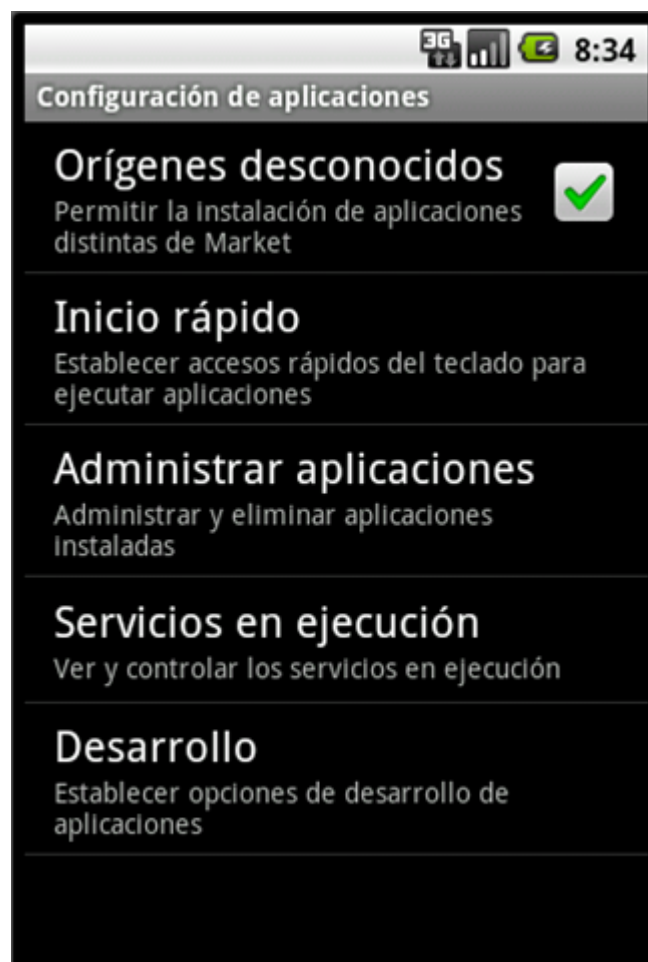


Ilustración 44: Instalar aplicación Android.

- Una vez activado lo mencionado en la ruta anterior, hay que copiar el fichero en la tarjeta de memoria del dispositivo seleccionamos el fichero y pulsamos en Instalar.

Una vez instalada la aplicación, para ejecutarla hay que ir al menú principal del teléfono Android y pulsar sobre una de las dos aplicaciones instaladas. Dependiendo de cada una de las aplicaciones, al ejecutarla pedirá un usuario y una contraseña, para empezar a usar cada una de las aplicaciones. Por defecto cada una de estas aplicaciones viene con usuarios y contraseñas por defecto las cuales se mencionan a continuación:

- **Aplicación Administrador:**
  - **Usuario:** admin
  - **Contraseña:** 100012345
- **Aplicación Cliente:**
  - **Usuario:** 100060967
  - **Contraseña:** 100060967

## 2. Servidor IIS

Aunque no se incluye el servidor en el sistema, se va a realizar una breve explicación sobre como instalar un servidor IIS, similar al que se ha usado al desarrollar el sistema.

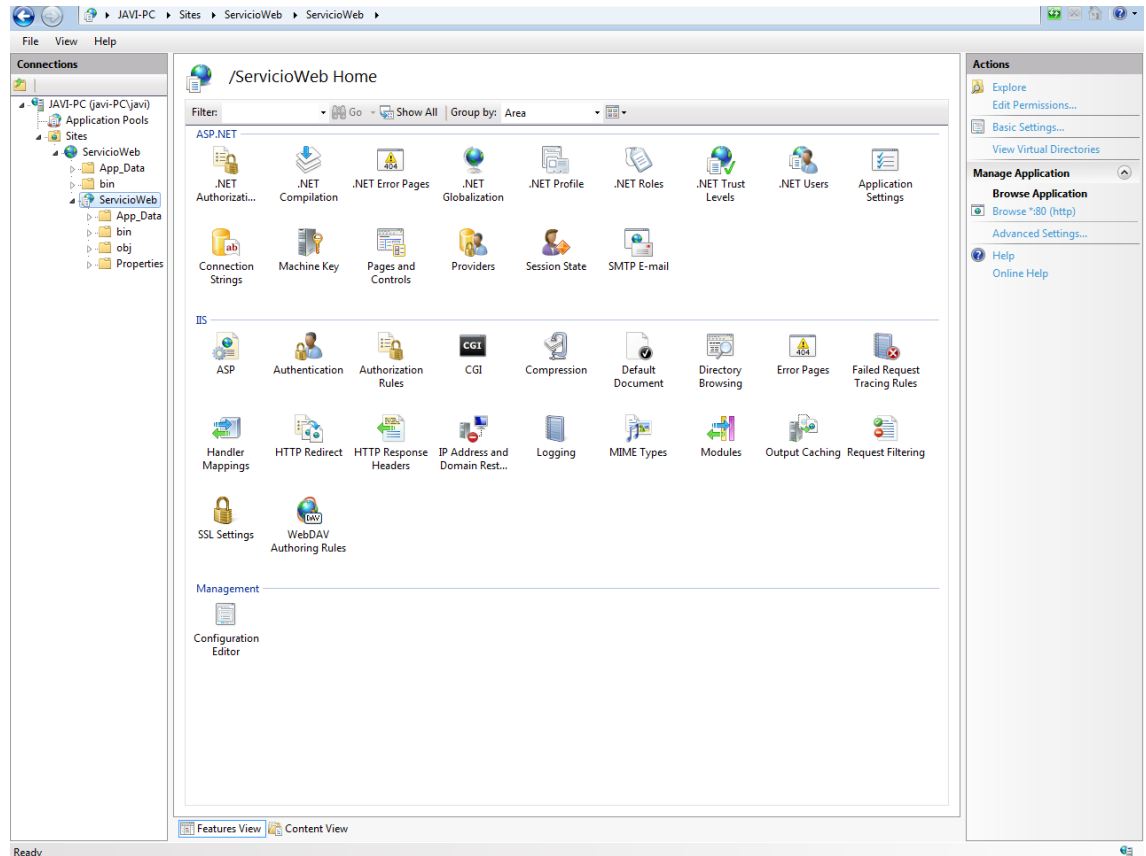
Para instalar un servidor IIS es necesario disponer de un sistema operativo Windows XP o superior, recomendando tener instalado Windows 7.

- Para instalar el servidor hay que hacer click sobre:

**Botón Inicio > Panel de Control > Programas > activar o desactivar las características de Windows.**

- Una vez en abierta la pantalla de características de Windows, hay que ir a *Internet Information Server* y pulsar en *Aceptar*.





**Ilustración 45: Servidor IIS.**

Simplemente con estos breves pasos disponemos de un servidor IIS, debido a que al instalar uno de los sistemas operativos de Windows, este viene de forma pre-instalada. Para configurar el servidor de forma que se opere con el servicio web creado bastaría con seguir la instalación del siguiente punto. Al instalar el servicio web a través de Visual Studio, este debe tener una aplicación raíz, que será donde se localice en el servidor IIS a través de su estructura de ficheros como se puede ver en la figura anterior.

### 3. Aplicaciones .NET

Para poder hacer uso de las aplicaciones creadas en .NET tales como el servicio web, la base de datos o el servicio de correo, es necesario disponer del IDE de desarrollo Visual Studio 2010. Para instalarlo hay que seguir los siguientes pasos.

- Descargar la última versión de la plataforma MSDN: <http://msdn.microsoft.com/es-es/vstudio/bb984878.aspx>
- Una vez descargado ejecutar el instalador y esperar a que se instale.

Una vez obtenido el entorno de desarrollo de Visual Studio 2010, ejecutamos la solución de .NET adjuntada con permisos de administrador.

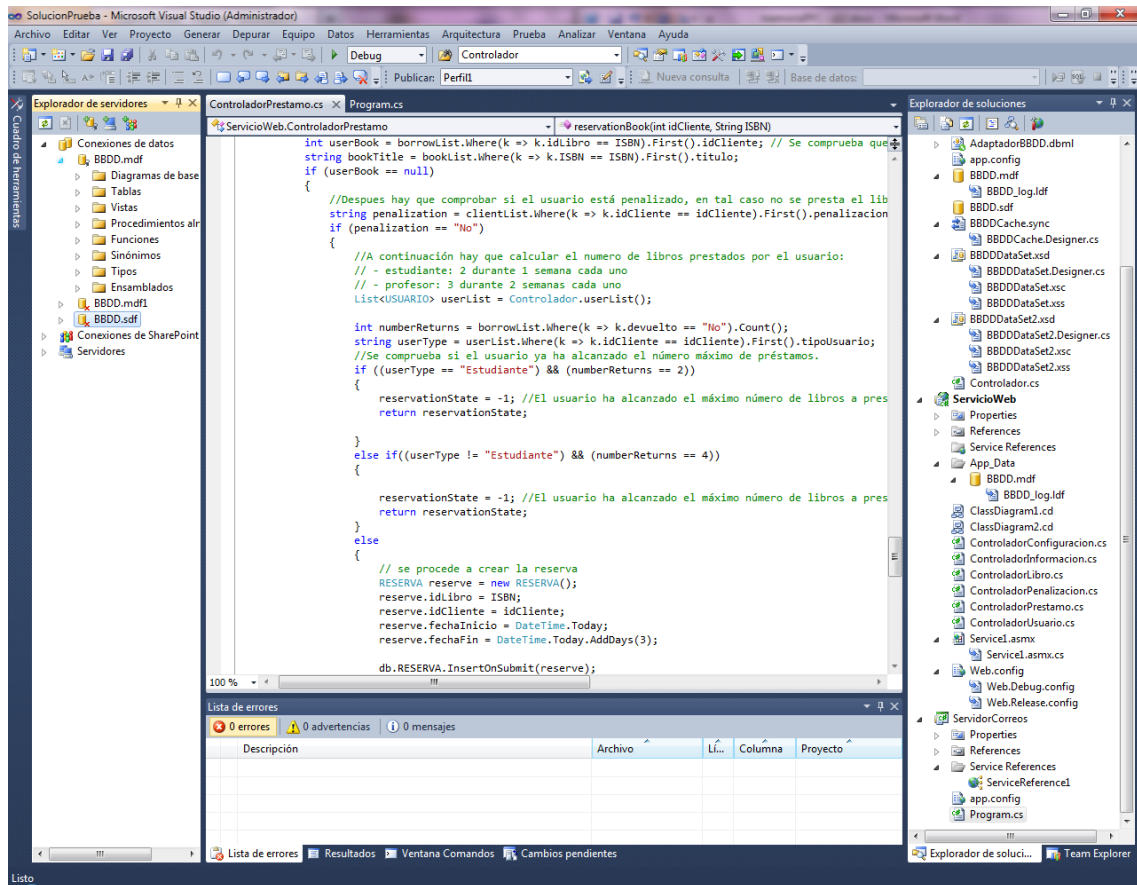


Ilustración 46: Solución Visual Studio 2010.

Se debe de obtener algo similar a lo mostrado en la figura anterior. Una vez ejecutado, hay que pulsar con el botón secundario del ratón sobre el proyecto llamado **ServicioWeb** y hacer click en publicar para poder lanzar el servicio web.

# Anexo D: Material entregado



En este apartado se van a describir que componentes se va a entregar al finalizar el proyecto fin de carrera con objeto de aclarar cual va a ser el contenido que se va a entregar junto con esta documentación. Hay tres directorios los cuales contendrán, las aplicaciones Android, la solución de .NET y la memoria del proyecto fin de carrera.

## **1. Aplicaciones Android**

Se va a hacer entrega de aplicaciones Android destinadas al manejo de los clientes. En total se entregan dos aplicaciones cuyo objetivo es el de ser usadas por distintos usuarios finales. Estas aplicaciones son las que se citan a continuación:

### **1.1 Aplicación Android administrador**

Esta destinada para los administradores del sistema, que supuestamente serán los bibliotecarios. Se entrega un fichero con extensión .apk llamado **AdministradorApp.apk**

### **1.2 Aplicación Android usuario**

Esta destinada para los usuarios finales del sistema, que supuestamente va a ser a los que se presten los libros de la biblioteca. Se entrega un fichero con extensión .apk llamado **UsuarioApp.apk**.

## **2. Solución .NET**

Se hace entrega de la solución .NET que contiene el Servicio Web, así como la Base de datos y la solución que permite enviar correos a través del servidor.

### **1.1 Servicio Web**

Se componen de una serie de proyectos de .NET Framework 3.5. Para ejecutar el servicio web el usuario debe tener instalado el IDE de desarrollo Visual Studio 2010.

### **1.2 Base de datos**

Esta contenida dentro de la solución de Visual Studio entregada. Únicamente hay que tener instalado el IDE de desarrollo Visual Studio 2010.

### **1.3 Servidor de correo**

Esta contenida dentro de la solución de Visual Studio entregada. Únicamente hay que tener instalado el IDE de desarrollo Visual Studio 2010.

## **3. Memoria**

Consiste en la entrega en formato .pdf del presente documento.